

**TITLE:**

Model and Control of an  
Autonomous Robot Dog.

**THEME:**

Final Thesis.

**PROJECT PERIOD:**

10<sup>th</sup> semester,  
February 1<sup>st</sup> - June 7<sup>th</sup> 2007

**PROJECT GROUP:**

1037d

**GROUP MEMBERS:**

Mikael Berg Andersen

**SUPERVISOR:**

Prof. Jakob Stoustrup

Lektor Jan Helbo

**NUMBER PRINTED:** 4

**NUMBER OF PAGES:** 170

**FINISHED:** June 7<sup>th</sup> 2007

**ABSTRACT:**

This project concerns the modeling and control of an autonomous quadruped robot, and aims toward achieving a smooth walk without falling. For this there were constructed a model, path planner, steering control, trajectory planner and supervisor.

The system was first analyzed, then trot locomotion was decided upon, with a ballistic curve movement for the feet. The balance of the robot was found to be dynamic. The falls were chosen to use to give the forward motion of the walk; therefore the angle of the body was controlled inside a limited circle. The steering controller was constructed as a linear ratio controller.

The model was divided into manipulator kinematics, inverse manipulator kinematics and manipulator dynamics, which were modeled separately, then combined to form one complete model. The model was structurally verified both with constant and variable inputs, where it proved the general properties of the model.

The path planner and trajectory planner, controls where the legs are moved to each time. The supervisor secures that the phase is kept synchronous between the different control parts.

The constructed system was tested both with, and without noise on the feedback signals, and the control system proved to be able to keep the rotations of the body inside the limited circle. The results of the test were visualized in a 3D-environment, to demonstrate the real system.



# PREFACE

This report is written by Mikael Berg Andersen from the Department of Control Engineering, Aalborg University (AAU). The theme for the 10<sup>th</sup> semester at Intelligent Autonomous Systems (IAS) is *Master Thesis*. Under these a project in making an autonomous robot dog walk was undertaken.

The report documents the work carried out on the above mentioned project in the first part of 2007, and is specifically intended for the supervisors, the examiner, IAS students and other individuals interested in the modeling and control of a robot dog.

Aalborg University, Denmark, 2007

---

Mikael Berg Andersen

---

## READING INSTRUCTIONS

The report begins with an introduction where the rest of the report are outlined. The report is overall consists of four parts; *Analysis*, *Modelling*, *Control* and *Closure*. Supplementary material which supports the various chapters is located in the *Appendix*.

In this report references are carried out according to the Harvard-method, e.g. [Cra05] or [Cra05, p. 100] if a more precise reference to e.g. a page or equation is needed. The full reference list can be located on page 131. Equations are numbered continuously throughout the report in the form of (9.1), while tables and figures are in the form of 9.1.

Abbreviations are written in their full length the first time, followed by the abbreviation itself in parentheses. The full abbreviation list can be located on page ix.

The nomenclature lists can be found on page vii.

The coordinate systems there will be used in this report are all right hand coordinate systems.

A CD-ROM containing project relevant material such as Simulink<sup>®</sup> models, MATLAB<sup>®</sup> code, papers, Gsim, etc. it can be found at the end of the report. If e.g. a manipulator kinematics model MATLAB<sup>®</sup> file is referred to, it is done in the following way, `/simulations/model/kinematics.m`, meaning that the file is located in this location on the CD-ROM. The folder structure is demonstrated below:

- Simulations
  - Control
  - Model
  - Verification
- Sources
- Video

# TABLE OF CONTENTS

<b>Nomenclature</b>	<b>vii</b>
<b>Abbreviations</b>	<b>ix</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 Problem Formulation</b>	<b>7</b>
<b>3 Newcastle robot dog</b>	<b>9</b>
<b>4 Quadruped Robot Walking</b>	<b>17</b>
4.1 Fall analysis . . . . .	23
<b>5 Central Pattern Generator</b>	<b>31</b>
<b>6 Visualization</b>	<b>37</b>
<b>7 Control System Architecture</b>	<b>41</b>
<b>8 Control System Objectives</b>	<b>45</b>
<b>9 Manipulator Kinematics Model</b>	<b>47</b>
<b>10 Inverse Manipulator Kinematics Model</b>	<b>55</b>
<b>11 Manipulator Dynamics Model</b>	<b>61</b>
<b>12 Structural Model Test</b>	<b>67</b>
<b>13 Sensor Modeling</b>	<b>73</b>
13.1 Rotational Dynamics . . . . .	73
<b>14 Trajectory Planner</b>	<b>79</b>
14.1 Walking . . . . .	80

## TABLE OF CONTENTS

---

14.2 Start Walk . . . . .	84
14.3 Test . . . . .	85
<b>15 Path Planner</b>	<b>89</b>
<b>16 Steering Control</b>	<b>97</b>
<b>17 Supervisor</b>	<b>109</b>
<b>18 System test</b>	<b>113</b>
18.1 Period time . . . . .	114
18.2 Stability . . . . .	115
18.3 Stability with White noise . . . . .	121
<b>19 Conclusion</b>	<b>127</b>
<b>20 Perspectives</b>	<b>129</b>
<b>Bibliography</b>	<b>131</b>
<b>A Manipulator Kinematics</b>	<b>135</b>
<b>B Inverse Manipulator Kinematics</b>	<b>137</b>
<b>C Inertia tensors</b>	<b>143</b>
<b>D Manipulator Dynamics</b>	<b>145</b>
D.1 Distribution of Gravity force . . . . .	145
D.2 State-space equations . . . . .	149
D.3 Structural Test . . . . .	150
<b>E Structural Test of Model</b>	<b>153</b>

# NOMENCLATURE

Parameter	Description	Unit
$a$	Acceleration	[m/sec <sup>2</sup> ]
$A_B$	A discrete signal from the steering controller send when $\theta_{N,m}$ is larger then $\theta_D$ , the signal can have the values; 1 avoiding fall, 2 fall is avoided, 0 else	[-]
$A_D$	A discrete signal from the supervisor to start the walk, the signal can have the values; 1 for start, 0 else	[-]
$a_i$	The distance from $\hat{Z}_i$ to $\hat{Z}_{i+1}$ measured along $\hat{X}_i$	[m]
$d_i$	The distance from ${}^i\hat{X}_{i+1}$ to $\hat{X}_{i+1}$ measured along ${}^i\hat{Z}_{i+1}$	[m]
${}_i^if$	The force required in the $i$ element	[N]
$F$	Force	[N]
$F_D$	A discrete signal from the supervisor to tell which phase the walk is in, the signal can have the values; 1 start walk by lifting the feet of LP2 (LF and RR) from the floor, 2 walk with LP2 in swing phase, 3 walk with LP1 (RF and LR) in swing phase	[-]
$H$	The height to the floor along the z axis of coordinate system {1}	[m]
$L$	The length	[m]
$m$	mass	[kg]
${}^{i+1}N_{i+1}$	The moment acting om frame $i + 1$	[Nm]
${}_i^n$	The torque extended on link $i$ by link $i - 1$	[Nm]
${}^{i-1}_iP$	The position vector from $\{i - 1\}$ to $\{i\}$	[m]
$\bar{P}$	Position vector, the submerged index will contain the name	[m]
$r$	radius	[m]
$R$	Ratio for correction of the feet	[-]
${}^{i-1}_iR$	Rotation matrix from $i$ to $i - 1$	[-]
$t$	The time in the period	[sec]
$t_d$	The period time	[sec]
$t_s$	The time step size	[sec]
${}^{i-1}_iT(\theta_i)$	Transformation matrix from $i$ to $i - 1$	[-]
${}^{i-1}\hat{X}_i, {}^{i-1}\hat{Y}_i, {}^{i-1}\hat{Z}_i$	Unit vectors X, Y and Z respectively to $\{i\}$ , expressed in $\{i - 1\}$	[m]
$V$	Velocity	[m/sec]

---

Parameter	Description	Unit
$B$	Submerged index for the steering controller	[-]
$CM$	Submerged index length or angle to the CM	[-]
$D$	Submerged index for the supervisor	[-]
$f$	Submerged index for fall	[-]
$g$	Submerged index for the gravity	[-]
$hit$	Submerged index for a hit	[-]
$IP$	Submerged index for inverted pressure	[-]
$j$	Submerged index for the joint	[-]
$l$	Submerged index for against supporting leg	[-]
$LF, RF, LR, RR$	Submerged index for the LF, RF, LR and RR leg, shoulder or foot, respectively	[-]
$m$	Submerged index for measured value	[-]
$N$	Submerged index for NBT	[-]
$opp, sid$	Submerged index for opposite and the side falling toward, respectively	[-]
$P$	Submerged index for the path planner	[-]
$Sw$	Submerged index for the swing phase	[-]
$T$	Submerged index for the trajectory planner	[-]
$x, y, z$	Submerged index for the x, y and z axis, respectively	[-]
$i, f$	Submerged index for start part and end part, respectively	[-]
$\alpha$	Angular acceleration	[rad/sec <sup>2</sup> ]
$\alpha_i$	The angle between $\hat{Z}_i$ to $\hat{Z}_{i+1}$ measured about $\hat{X}_i$	[rad]
$\theta_i$	The angle between $\hat{X}_i - 1$ to $\hat{X}_i$ measured along $\hat{Z}_i$	[rad]
$\theta$	The angle	[rad]
$\dot{\theta}$	The angular velocity	[rad/sec]
$\ddot{\theta}$	The angular acceleration	[rad/sec <sup>2</sup> ]
$\theta_D$	The threshold angle for the body, given of the supervisor	[rad]
$\tau$	The torque	[Nm]
$\rho$	Body angle to give resulting torque to stop fall	[rad]
$\vartheta_{CM}$	The angle from vertical to the line from the supporting feet to the CM.	[rad]
$\psi$	The angle measured around the shoulder of the rear supporting leg, from a line parallel with the back of the body to a line to the point defined by the submerged index.	[rad]
$\omega$	The angular velocity	[rad/sec]

---



# ABBREVIATIONS

**AAU** Aalborg University

**AIBO** Artificial Intelligence roBOt

**AM** Adaptive Module

**CM** Center of Mass

**CPG** Central Pattern Generator

**CTM** Coordinate Transformation Matrix

**GSim** Graphical SIMulation

**IT** Inertia Tensor

**IAS** Intelligent Autonomous Systems

**LF** Left Front leg

**LL** Lower Leg

**LP** Leg Pair

**LP1** Leg Pair 1

**LP2** Leg Pair 2

**LP3** Leg Pair 3

**LR** Left Rear leg

**NBT** Newcastle Bull Terrier

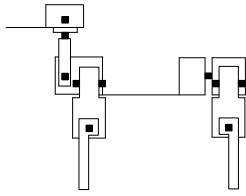
**RAT** Right Angle Triangle

**RF** Right Front leg

**RR** Right Rear leg

**UL** Upper Leg





# INTRODUCTION

Robotic technology has over the last several decades been widely used for manufacturing applications. As the technology became cheaper, it became economically viable to use it for non-manufacturing purposes. In correlation with this, SONY<sup>®</sup> developed the Artificial Intelligence roBOt (AIBO), which is an autonomous robot dog. Figure 1.1 shows the AIBO ERS-7M3 seen from its right side. A RoboCup<sup>®</sup> soccer league competition was started to increase research in robotics, with the overall goal of developing a team of humanoid robots to win the official human World Soccer Championship in year 2050.



FIGURE 1.1: *An illustration of the AIBO ERS-7M3 from SONY<sup>®</sup>.*

The SONY<sup>®</sup> AIBO's are used as players in the Four-Legged League of the RoboCup<sup>®</sup> competition. In this league a team consists of four SONY<sup>®</sup> AIBO's. A match consists of two teams playing on a  $4 \times 6$  m field, where the robot dogs work autonomously without any external influence. The field is illustrated on Figure 1.2. [Com]

As in human soccer there is some close contact when the players are fighting for the ball. This contact can get the robot dogs to fall over when they are walking on their feet, to minimize this the dogs in allot of the teams are crawling on their knees because it is lowering the Center of Mass (CM) and thereby making the dog more stable, but also decreasing its obtainable velocity in forward direction.

Each year a world cup championship for the RoboCup<sup>®</sup> takes place. Victory in the Four-Legged League gives much prestige, therefore many universities participate. Un-

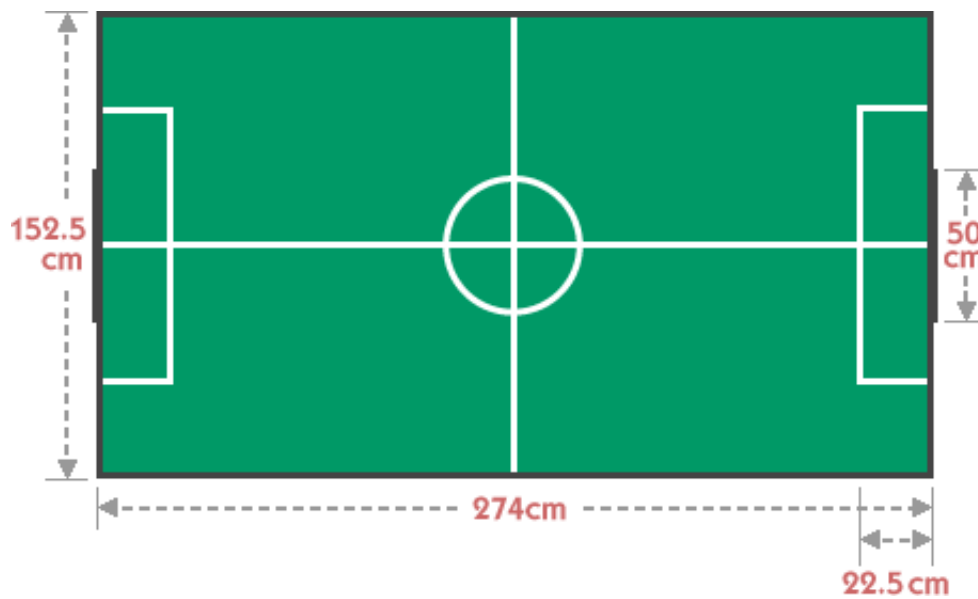


FIGURE 1.2: An illustration of the field used in the Four-Legged League.

fortunately SONY<sup>®</sup> announced that production of the AIBO was to be discontinued as of March 2006 [Eur]. Because of this the RoboCup<sup>®</sup> has began looking to find an alternative robot dog to replace the SONY<sup>®</sup> AIBO. There are several different organizations competing to get their dog selected to be the new dog type in the Four-Legged League. The organization behind the chosen dog type will then earn money by selling the new robot dog specifically for the Four-Legged League. All the teams that are participating will then have to buy a minimum of four dogs. The University of Newcastle is one of these organizations attempting to have their design accepted as the new standard platform for the Four-Legged League. This platform will be the basics for this project. Unfortunately it is not at disposal, therefore the verifications will be made with 3D visualizations. These will be made with Graphical SIMulation (GSim), which is illustrated on Figure 1.3.

Fast movement over the field is a key factor in human soccer, therefore is it desirable also to use in the Four-Legged League. This requires a controller to correct the movements when disturbed for instant by contact with another player or an object on the field. The movement over the field has to be with a speed where there is time to correct for falls. It also requires generation of walking trajectories for both the movement of the feet and the phase between the legs.

When the robot dogs are constructed they can have minor differences, this will influence the model and control of it, therefore it is desirable to correct the model for each dog. Animals uses a Central Pattern Generator (CPG) to update their model and to generate reference signals for the muscles.

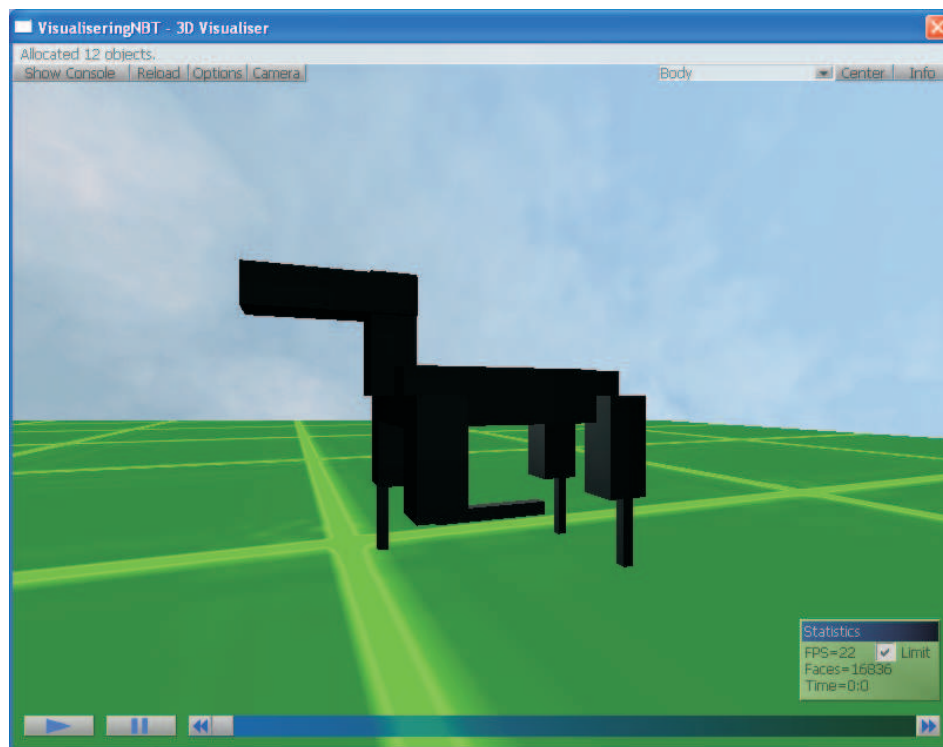


FIGURE 1.3: Illustration of view in GSim, which will be used to verify the final system.

The different parts of this project and the connections between them are illustrated on Figure 1.4. The white boxes illustrate the different parts of the project and the arrows the connections between them. The output from each part includes all the inputs it receives. The control system architecture and control system objectives are not included on the figure because they receive the output from all the parts in the analysis and declare the goals for the rest of the project. Therefore they would have connections to all parts in the model and control which would make the figure incalculable.

The content of the different parts of the project will shortly be described in the following:

**Analysis of Gait:** Analysis of different walking locomotion and choice of the best suited one. It also includes an analysis of how a leg should be moved to obtain the most optimal path, and when the foot should touch the ground. This part can be found in Chapter 4 on page 17.

**Analysis of NBT:** Describes the prototype of the standard platform designed at the University of Newcastle. This includes descriptions of sizes, weights, sensors, how there will be referred to the different parts, etc. This part can be found in Chapter 3 on page 9.

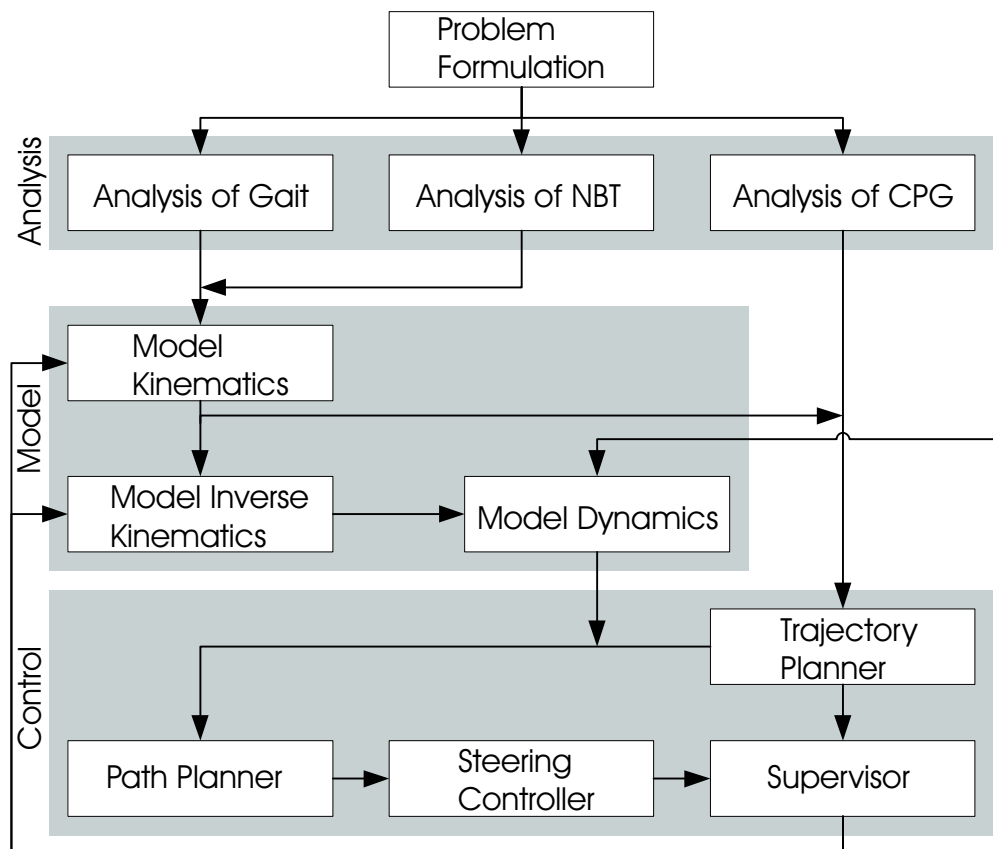


FIGURE 1.4: Illustration of the connection between the different parts of the project.

**Analysis of CPG:** Analysis overall what a CPG does and how it can be used in the trajectory planning and adaptation of CPG. It can be found in Chapter 5.

**Manipulator Kinematics Model:** Designs the references between the different parts of the prototype, this will end with a way to calculate between the different parts of it. This can be found in Chapter 9 on page 47.

**Inverse Manipulator Kinematics Model :** Designs a way to calculate back wards in the Kinematics model. It can be found in Chapter 10 on page 55.

**Manipulator Dynamics Model:** Designs a model of the forces acting on the prototype and the forces needed to make a movement. This will make it possibly to make the prototype move. This can be found in Chapter 11 on page 61.

**Trajectory planner:** Defines how the legs shall be moved, with the walking locomotion found in analysis of gait. This include position of the trajectory for the individual foot. It can be found in Chapter 14 on page 79.

**Path planner:** Designs a control to hold the phase between the legs. It can be found in Chapter 15 on page 89.

**Steering Controller:** Designs control to prevent the robot dog from falling, this reactions will include modifying on the trajectories from the path planner. It can be found in Chapter 16 on page 97.

**Supervisor:** Designs a supervisor to keep the control system synchronous. It can be found in Chapter 17 on page 109.

The different blocks in the model and control fields will be verified in the end of their design, the model further will be tested together. The control system objectives, system test, conclusion and perspectives have also been disregarded on the figure because they also would make the figure incalculable. The verifications and test parts will be visualized in a 3D game engine, this will be described in a part called Visualization. Since the visualization will be made with a 3D game engine and Simulink®, it is necessary to calculate the sensor signals, therefore will there be made a sensor modeling chapter. The control system architecture, control system objectives, sensor model, visualization, model- and control verification will shortly be described in the following

**Control System Architecture:** Combining the control system from the analysed parts in the analysis. This can be found in Chapter 7 on page 41.

**Control System Objectives:** States the goals for the project. This can be found in Chapter 8 on page 45.

**Sensor Modeling:** Calculates the sensor signals there are used by other parts of this project. This can be found in Chapter 13 on page 73.

---

**Visualization:** Analysis of the visualization tool and description of how the map is changed.

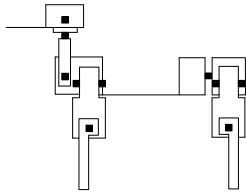
The implementation of objects to visualize the prototype of the robot dog will also be described. All this can be found in Chapter 6 on page 37.

**Model Verification:** Tests the model for if it fulfills the specifications from the objectives.

This can be found in Chapter 12 on page 67 and appendix E on page 153.

**System test:** Tests that the control fulfills the claims specified in the objectives. It can be found in Chapter 18 on page 113.





# 2

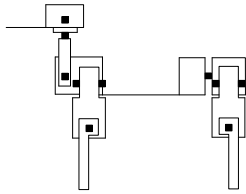
## PROBLEM FORMULATION

With background in the problems described in the introduction, the problem that will be investigated is as follows:

*How can it be avoided, that the new robot dog falls when it is walking on a plain surface, or when it comes in contact with another object?*

The main goal of the project is to make the robot dog walk fast, therefore the walk must be stable. The walk will be constrained to a straight line, turning will be disregarded.





## NEWCASTLE ROBOT DOG

*This chapter analysis a prototype for one of the platforms suggested to become the new standard platform for the Four-Legged League, this platform is designed at the University of Newcastle.*

The prototype has to fulfill some design claims. The new robot dog has to be designed with a joint placement similar to that of the AIBO. This will benefit the teams in the Four-Legged League as they will be able to capitalize on their experiences from earlier years. Furthermore it will also allow the dog to continue playing with the official ball for the league. The new dog shall also have a look as appealing as the AIBO. To meet these three criteria, the Newcastle design team has used a bull terrier as inspiration. A drawing of a bull terrier is illustrated in Figure 3.1. This dog race is chosen because it almost is the same size as the AIBO, and it is possible to implement the joints, without getting the dog to look abnormal. The dog is not named yet, but will be referred to as the Newcastle Bull Terrier (NBT) in the sequel.

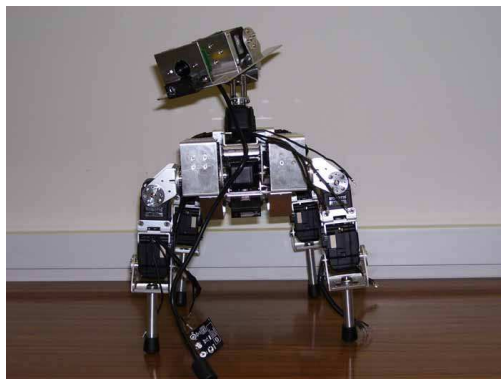


FIGURE 3.1: An illustration of a bull terrier.

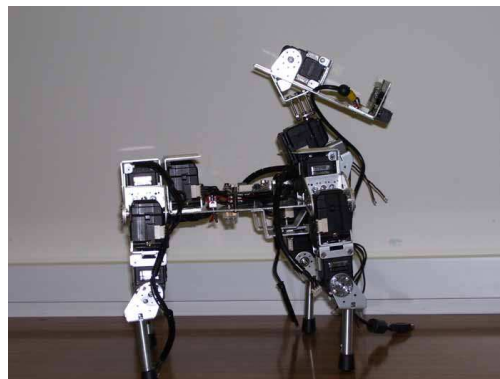
The final design and construction of the NBT is not yet complete. However there is a prototype, which is the focus of this project. The prototype there will be used is of July 27<sup>st</sup> 2006. Figure 3.2 on the following page shows four pictures of the prototype for the

---

NBT, seen from different angles. Figure 3.2(a) shows the NBT from the front, Figure 3.2(b) is seen from its right side, Figure 3.2(c) is seen from behind and Figure 3.2(d) is seen from the left front side. The prototype shown in the images will be covered by a body, to give the dog an appealing look. This body is not yet constructed.



(a) The NBT seen from the front.



(b) The NBT seen from the right side.



(c) The NBT seen from behind.



(d) THE NBT seen from the front right side.

**FIGURE 3.2:** *Images of the NBT seen from four different angles. These pictures are from July 27<sup>st</sup> 2006.*

The prototype of the NBT is depicted as three schematics seen from different angles on Figure 3.3. These schematics illustrate how the various body parts of the NBT are connected. Figure 3.3(a) is seen from the left side, and Figure 3.3(b) and 3.3(c) are seen from above. Figure 3.3(c) shows the dog lying on its stomach, with its legs perpendicular to the body, to show all the joints in one figure.

The black squares in the schematics illustrate the joints (motor axis) and where they can be rotated. The direction of the rotation depends on the joint layout, as described below:

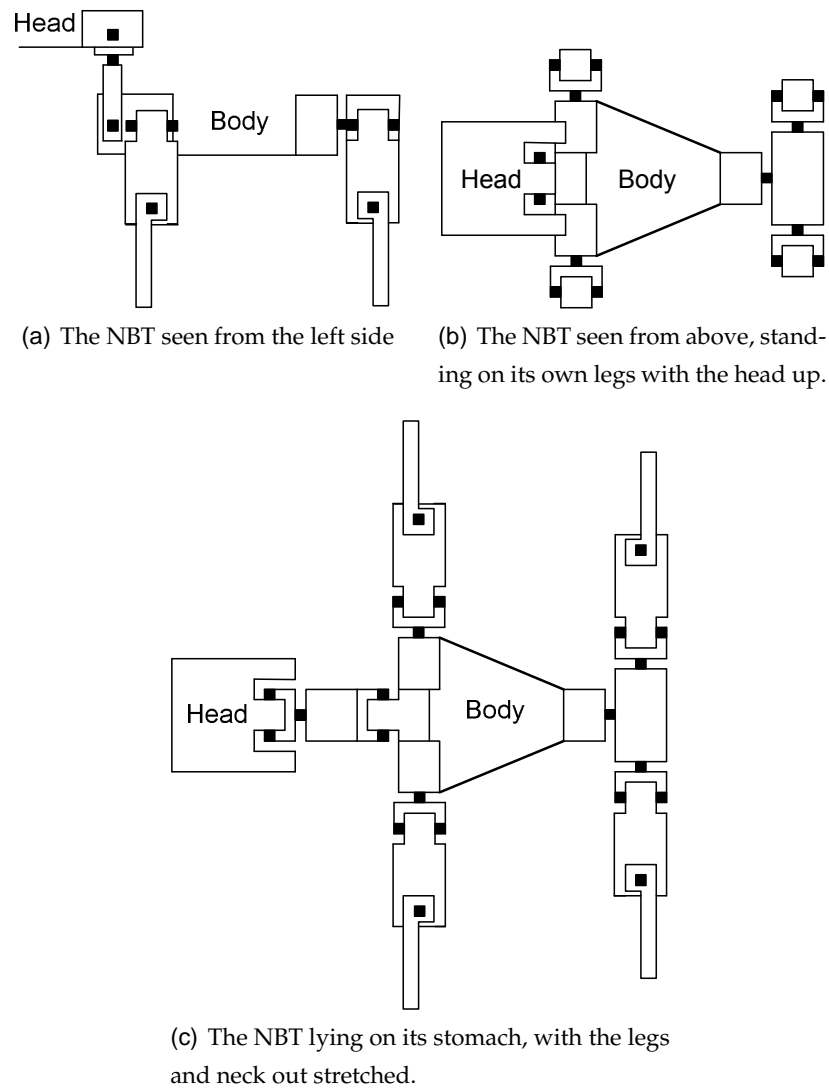
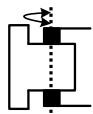
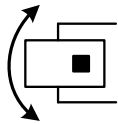


FIGURE 3.3: Schematics of the NBT depicted in three different positions.



This joint rotates around the two black dots. The dotted line illustrates the axis of rotation.

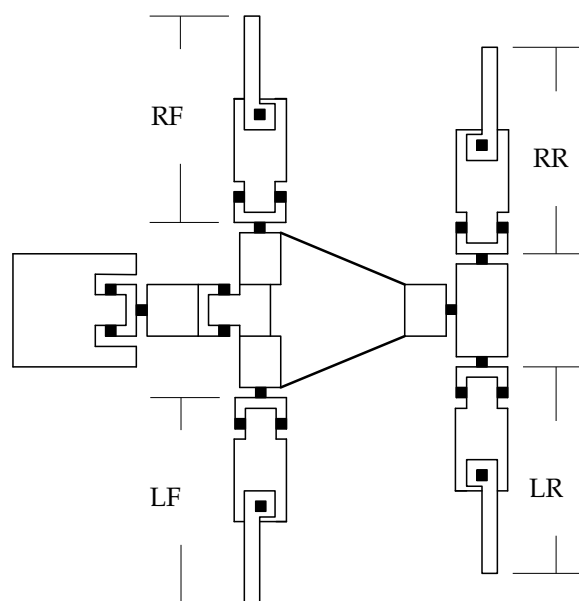


This joint rotates around the black dot. It rotates in the plane of the paper, as illustrated by the arrow.



This joint type rotates around the black dot. A dotted line illustrates the axis of rotation.

The central part of the NBT is the body, to which the neck and legs are mounted. The legs will be referred to as follows; Left Front leg (LF), Right Front leg (RF), Left Rear leg (LR) and Right Rear leg (RR). These notations are illustrated on Figure 3.4.



**FIGURE 3.4:** *Illustration showing the notations for the legs. Viewed from above the NBT.*

Each of the legs consist of a Lower Leg (LL) and Upper Leg (UL) connected by a knee. The NBT do not have feet, the legs end in round tubes, with slip resistant surface. A leg is illustrated in Figure 3.5.

The legs are connected to the body by shoulders, composed of two degrees of freedom, giving each of the leg three degrees of freedom. The shoulders have a displacement

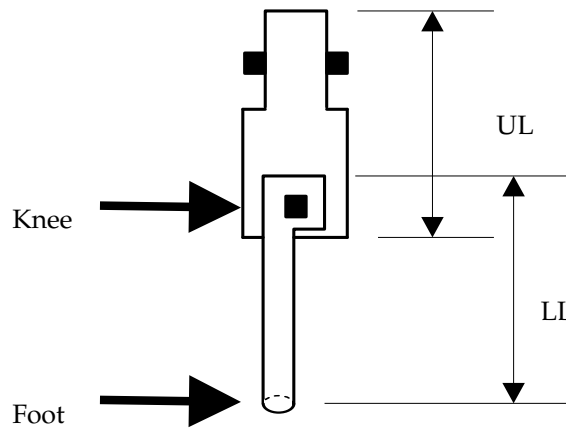


FIGURE 3.5: Illustration of the LF leg showing the notations for the different parts of the legs. Viewed from its right side.

between the two rotation directions of  $0.0110$  [m]. The knee also have a displacement of the same size, this is in forward direction for the front lags and in backwards direction for the rear legs. The displacements of a front leg are illustrated on Figure 3.6. The figure is made as a line drawing because it better demonstrates the displacement.

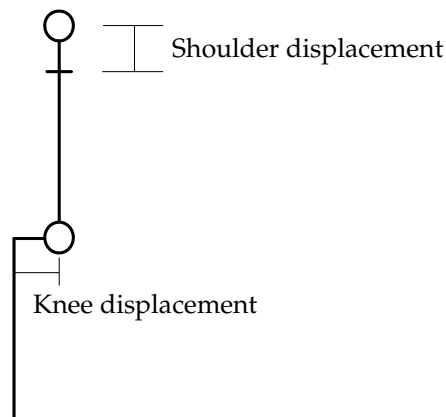


FIGURE 3.6: Illustration of the shoulder and knee displacement.

There also exists a joint between the front and rear shoulders, which will be referred to as the hip. It is implemented to make the rotation of the dogs back when it is walking. The AIBO did not have this joint. It has been implemented to provide the NBT with a smoother walk. The hip and shoulders are illustrated on Figure 3.7.

The NBT has a total of 16 degree of freedom, where 13 of these are used for walking, and three are used to position the head. Each of the 16 degrees of freedom are moved by

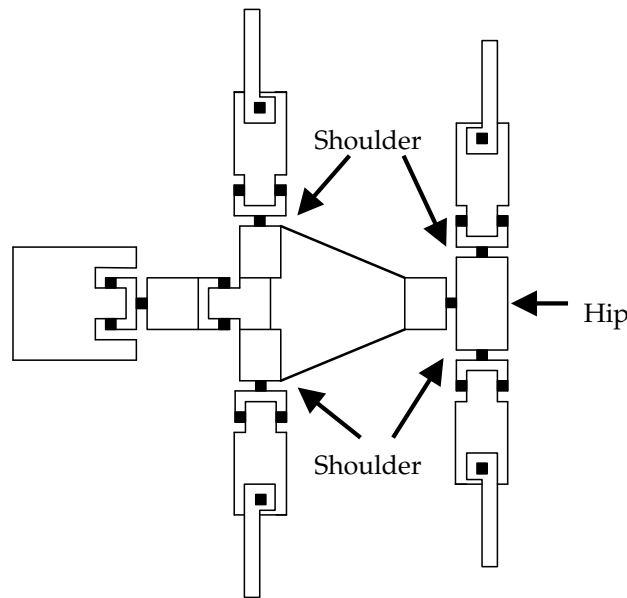


FIGURE 3.7: Illustration showing where the shoulders and hip of the NBT are.  
Viewed from above the NBT.

a motor. These motors are of the type DX-113 from the Dynamixel series. They have a working range of  $5.24$  [rad] ( $300^\circ$ ), and a mass of  $0.058$  [kg]. For more information about the motors see the importers homepage at [Tri].

The sizes and weights of the different parts of the NBT are shown in Table 3.1. The lengths are measured parallel with the longitudinal axis of the NBT, the widths are measured parallel to the latitudinal axis and the heights are measured vertically, when the NBT is standing on its feet, the axis are illustrated on Figure 3.8.

Part	Length	Width	Height	Weight
Body	$0.1700$ [m]	$0.1088/0.0368$ [m]	$0.0464$ [m]	$0.2900$ [kg]
Head	$0.1300$ [m]	$0.0500$ [m]	$0.0313$ [m]	$0.0580$ [kg]
Hip	$0.0360$ [m]	$0.0740$ [m]	$0.0464$ [m]	$0.1160$ [kg]
LL	$0.0100$ [m]	$0.0100$ [m]	$0.0866$ [m]	$0.0734$ [kg]
Neck	$0.0368$ [m]	$0.0368$ [m]	$0.1000$ [m]	$0.1160$ [kg]
UL	$0.0368$ [m]	$0.0368$ [m]	$0.0755$ [m]	$0.1160$ [kg]

TABLE 3.1: The sizes and weights of the different parts of the NBT. The first width of the body is in the front and the second is in the rear.

Figure 3.8(a) illustrates the NBT seen from its left side, and Figure 3.8(b) illustrates the NBT seen from above.

The size of the legs and neck are calculated from the center of the joint to the center of



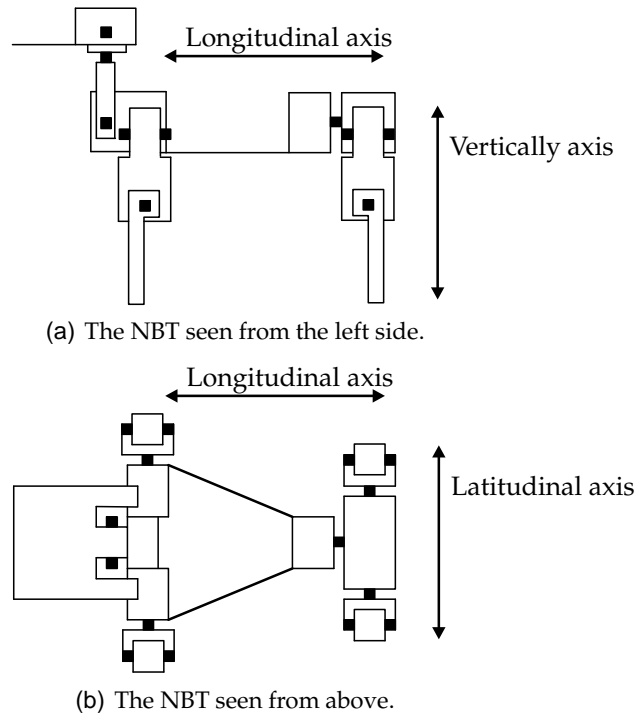


FIGURE 3.8: Schematics of the NBT illustrating the placement of the longitudinal-, latitudinal- and vertically axis.

the joint. The LL is displaced 0.0110 [m] forward for the front legs and backwards for the rear legs. The first width for the body is the front and the second is the rear. The weight of the LL is calculated from the weight of aluminum, and the rest is calculated from the weight of the motors. The total weight of the NBT is calculated to 1.3376 [kg].

There are implemented several sensors on the NBT, which are described below:

**Accelerometers:** Hitachi H48C Tri-Axis module. This provides acceleration in three directions, these are in the directions of the coordinate system axis's illustrated on Figure 3.9.

**Gyros:** ADXRS150EB. It is planed to mount three of these on the NBT. This will give the velocity in the three same directions as the accelerometers.

**Vision:** The actual camera is not yet decided, but at the moment there exists a Logitech PRO5000. There will be calculated directions and distances on the images imported from the camera.

**Distance sensors:** Sharp GP2D12 and GP2D120 mounted on the nose and chest, respectively. This will give the distance to objects in front of the NBT, one of these objects is usually the floor.

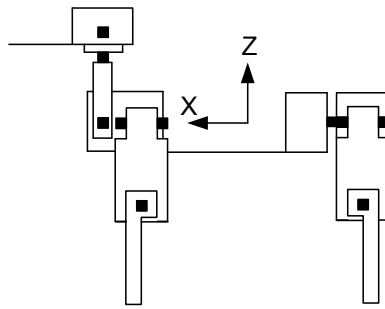


FIGURE 3.9: *Illustration of sensor placement on the NBT. The NBT is viewed from its left side.*

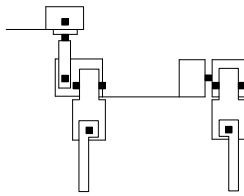
**Compass:** Horneywell hmc1055. This provides the 2D orientation of the NBT's body. This orientation is in the plan spanned by the X and Y axis of the coordinate system on Figure 3.9.

**Motors:** DX-113 from Tribotixs Dynamixel series. The motors have many feedback signals which can be used as sensors. The feedback signals are; position, temperature, load, input voltage, etc. The motor have a step size of  $0.0061[\text{rad}]$ .

The sensors will be interfaced to the controllers through a driver. This interface is not designed yet, therefore it will not be considered in the control system implementation. The signals to be used are the most convenient for the control because the drivers can be designed to fulfill this.

The University of Newcastle have a decision maker, vision processing and path planner from the AIBO, that will be used on the NBT, in a modified form. The decision maker chose the behavior of the NBT, this contains; Radio communication, movement speed, movement direction, etc. The vision processing recognizes objects on the field and the ball, this information is used by the decision maker. The path planner is planning the route that the NBT shall walk in to get to the position the decision maker want. These three parts are not at disposal in this project.

The NBT has two 400 [Mhz] processors installed. One of them is only used to the vision processing and the other one handles the rest of the computations. This means that there are a lot of processes running on the second processor, and only minimal is reserved for control, therefore the computations used by the control system has be limited. The processor power at disposal is not defined further from the design team at the University of Newcastle.



## QUADRUPED ROBOT WALKING

*This chapter, analysis different walks to find the most suited one for implementation. Two trajectories for the feet are also analyzed. The falling doing walk will be discussed at the end of the chapter, to address the problems regarding the tendency for the NBT to fall during walking.*

The locomotion can be accomplished by crawling on the knees or walking on the feet. The crawling lowers the CM, thereby giving more balance, but also slowing down the movement. Walking on the feet gives the possibility for faster movement because the legs are longer, however it also makes the construction more unstable, because the CM is higher. This can be eased by putting the legs at an angle to the body as illustrated on Figure 4.1(a). The higher the angle, the larger the standing area becomes, thereby giving a better balance. The standing area is the area between the feet, as illustrated on Figure 4.1(b), where the NBT is seen from above. The black dots illustrates the positions of the feet, and the gray square is the standing area. As described in Chapter 2 on page 7 is the main goal to make the NBT walk fast and keep the body stable. Therefore it is chosen that the NBT will walk on its feet with the legs at an angle to the body. The angle will be discussed after the gait, or method of walking, is chosen, which will be done in the following.

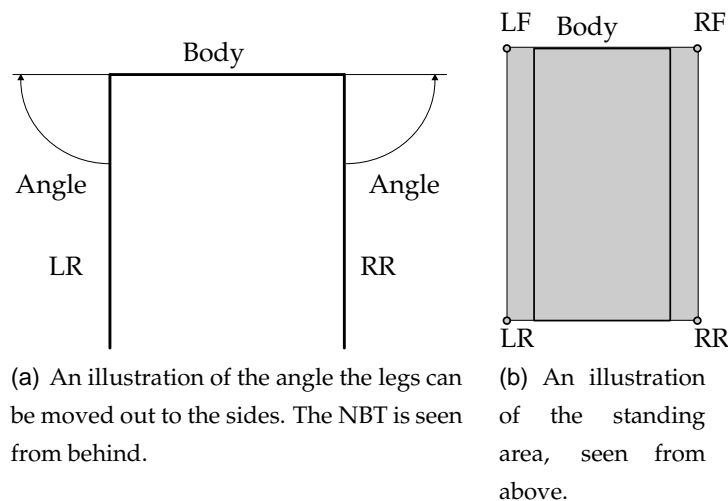


FIGURE 4.1: Illustrations of the leg placement for the NBT.

A quadruped legged construction can have different gaits, among these the following

are of interest; amble, bounce, curvet, gallop and trot. The problem formulation states that the body of the NBT shall be kept steady, this gets more difficult when the locomotion include flight sequences, therefore bounce and gallop are disregarded. Nature is usually using the optimal solution, therefore curvet locomotion is disregarded, because it is not a natural gait for a quadruped animal; this is also proved by other researchers [AMA03, p. 61]. The amble and trot gait will be described in the following:

**Amble:** The amble gait is cyclical and moves the legs in the right side together and the legs in the left side together. This is illustrated on Figure 4.2, where the legs joint together by the bar are moving together. The amble gait does not include flight phases, therefore it always has two feet on the ground and four when shifting which legs there are moving. Table 4.1 illustrates when the feet are touching the ground, in two walking cycles. A walking cycles consists of 4 stages, where the dashes symbolize when a foot is on the ground. [AMA03, p. 42-43] [JFK95, p. 697]

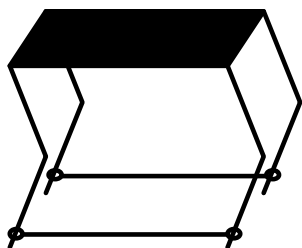


FIGURE 4.2: An illustration of amble gait.

Foot\State	Cycle 1				Cycle 2			
	1	2	3	4	1	2	3	4
LF	-	-		-	-	-		-
RF		-	-	-		-	-	-
LR	-	-		-	-	-		-
RR		-	-	-		-	-	-

TABLE 4.1: Movement sequence for amble walk.

A period of the walking sequence is illustrated on Figure 4.3, where the dotted legs are the LF and LR. The dotted legs move akin to each other as the other two also does.

This gait makes the body of the dog rotate around its own longitudinal axis, the axis is described in Chapter 3. The rotation of the body is around the feet it is standing on. When two of the legs are lifted from the ground, the body is rotated away from the moving legs to move the CM against the supporting legs and give the moving legs more clearing from the floor. Before the legs ends their forward motion is the body rotated back against them, so they can reach the ground again. The direction of rotation when the left legs are moving forward is illustrated on Figure 4.4.

Contact with other players can change the rotation of the body, because the contact momentary create a force in the NBT. This force will mostly be divided into a force

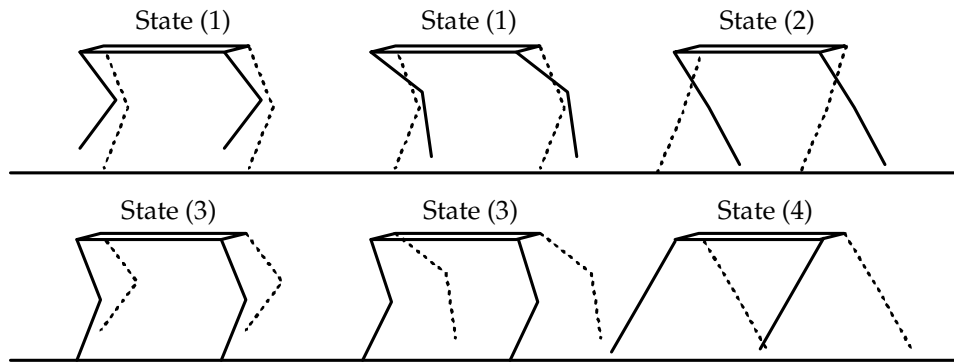


FIGURE 4.3: An illustration of a sequence of amble gait.

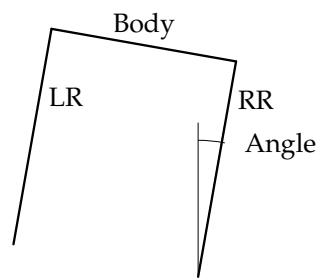
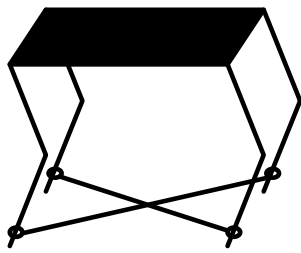


FIGURE 4.4: An illustration of the rotation of the body doing amble locomotion. The NBT is seen from behind.

in the rotation direction of the body and another direction. The force in the direction of the rotation, will increase or decrease the rotation of the body. Increasing of the rotation away from the moving legs can make the rotation to large, thereby giving problems with rotating back so the moving legs can be set down correctly. This gets more critical if the contact is just before the movement of the legs should have ended, because there is no time to react to the force. If the phase between the legs is out of synchronization, the legs the robot is standing on starts moving before the legs in forward motion are on the ground, thereby making the robot fall. If the hit makes a force in the direction from the standing legs to the moving legs it is also critical. In the start and end of the forward motion of the legs the force can result in a lag of rotation away from the ground, thereby the foot on one or both of the moving legs can get in contact with the floor. This can make the robot stumble.

**Trot:** The trot locomotion is also cyclical and moves the diagonal legs together, this means LF and RR together and RF and LR together. This is illustrated on Figure 4.5, where the joint legs are moving together. This locomotion does not include flight phases, and therefore it always has two feet on the ground and four when shifting legs. Two walking cycles for when the feet touches the floor are illustrated in

table 4.2. [AMA03, p. 43-44] [JFK95, p. 697]



	Cycle 1				Cycle 2			
Foot\State	1	2	3	4	1	2	3	4
LF	-	-		-	-	-		-
RF			-	-			-	-
LR			-	-			-	-
RR	-	-		-	-	-		-

FIGURE 4.5: An illustration of trot gait.

TABLE 4.2: Movement sequence for trot walk.

A period of the walking sequence is illustrated on Figure 4.6. Where the dotted legs again move akin to each other, which the other also does.

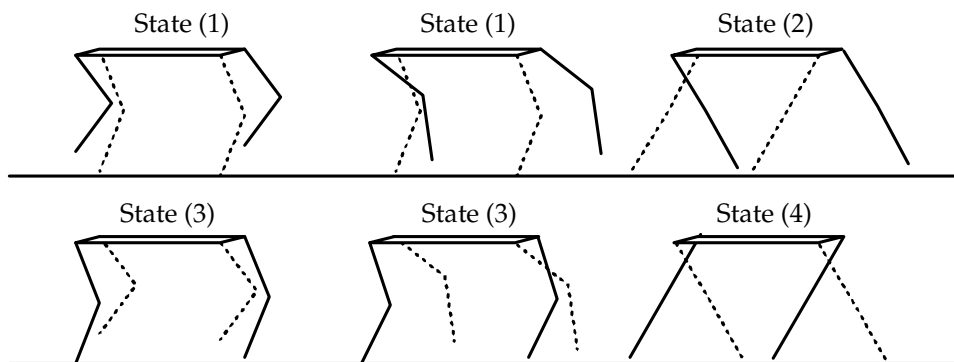


FIGURE 4.6: An illustration of a sequence of trot gait.

The body of a horse is always horizontal doing trot, which can be used as an initial condition in trot [AMA03, p. 43-44]. Contact with another player will still give a force resulting in a rotation of the body around the standing feet. This rotation can as earlier mentioned be minimized by moving the legs further away from the sides of the body, as illustrated on Figure 4.1.

Both amble and trot are usable locomotions, but one has to be chosen. Other researchers document that the amble gait is best for faster motion and the trot gait is best for slower motion [AMA03, p. 61]. They also document that trot is the most stable locomotion. This concurs with the analysis above, therefore is trot chosen because the body is kept more stable and balance is more important than speed. In later projects when the NBT can walk in trot with control of the balance, can it be interesting to look into amble locomotion.

The foot of a quadruped walking robot can be moved in different ways, of interest are ballistic and rectangular motions, seen from the floor. Both the sequences consisting of a swing- and stand phase. The legs are supporting the body in the stand phase while it is moved forward, as illustrated on Figure 4.7. The figure illustrates the supporting legs and body of the NBT. The points  $P1$  and  $P2$  are the same in the whole figure.

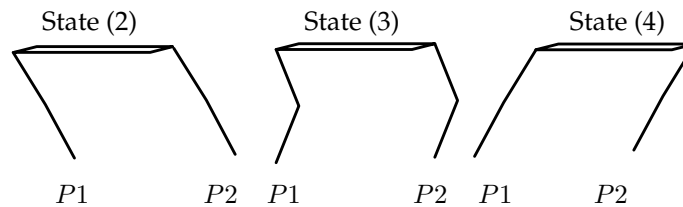


FIGURE 4.7: An illustration of how the body is moved forward over the standing feet.

The swing phase lifts the foot from the floor and moves it forward. At the end of the swing phase the foot's put down on the floor again. When the foot comes in contact with the floor a force will be generated, this impact force will give disturbances throughout the whole NBT dependent on how large the impact is. The swing- and stand phase are shifted between the legs as the trot locomotion defines. The ballistic and rectangular motions for the feet will be further described in the following:

**Ballistic:** Three periods of a ballistic movement of the foot, seen from the ground, is illustrated on Figure 4.8. The curve for the swing phase can be seen as an inverted parabola.

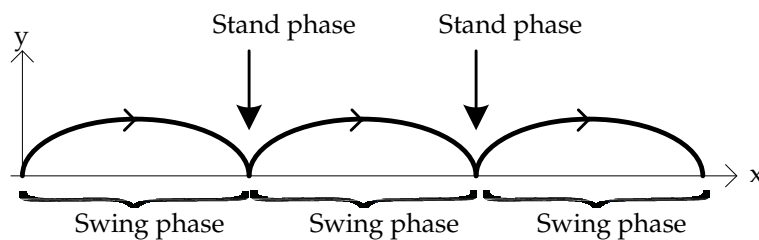


FIGURE 4.8: An illustration of the curve for ballistic movement the foot.

A period of the swing phase is illustrated on Figure 4.9.

The figure illustrates the swinging leg and the body.

**Rectangular:** One period of a rectangular movement of the foot, seen from the world, is illustrated on Figure 4.10. Dependent on how the rectangular is placed in correla-

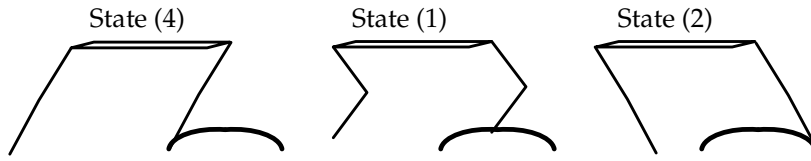


FIGURE 4.9: An illustration of the LF foot following the ballistic curve.

tion with the dog, the direction of the walk can be forward, backward, sideways etc. [BHS00]

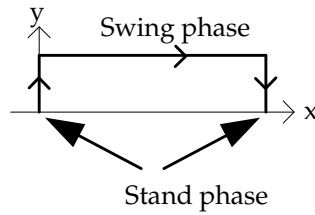


FIGURE 4.10: An illustration of the curve for rectangular movement of the foot.

The swing length along the ground and height the foot is lifted to clear the ground depends on the robot for both trajectories. The ballistic trajectory can have to lift the foot higher than the rectangular to clear the ground in the start and end of the forward motion. The rectangular motion has a longer trajectory than the ballistic in the forward motion, therefore it has to use more time on the swing phase than the ballistic motion. The source for the rectangular motion describes how it can be expanded to locomotion in other directions, this is not described for the ballistic trajectory, but is expected to be similar to the rectangular trajectory. The ballistic trajectory is expected to be the best suited one because the swing phase is the fastest. This gives the possibility for a faster locomotion without compromising the balance. The rectangular trajectory is furthermore having the disadvantage that the foot has to be stopped and started at each of the corners. The motion will be stopped at the end of the swing- and stand phase, this will reduce the effect of the impact with the ground to a minimum, therefore will it be disregarded.

The ballistic motion will be seen as an inverted parabola this gives Equation (4.1) for the swing phase.

$$y = \frac{-4h}{l^2}(x - hl)(x - (h + 1)l) \quad (4.1)$$

$y$  is the height the foot is lifted from the ground and  $x$  is the location in the movement as illustrated on Figure 4.11.  $h$  is the height of the curve in the middle and  $l$  is the length



of the curve.  $h$  and  $l$  depends on the angle the legs are moved forward, backwards and out to the side of the NBT's body, which depends on how the balance best is kept doing walk. The balance considerations are discussed in section 4.1.

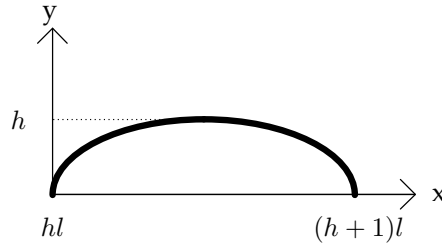


FIGURE 4.11: An illustration of the curve for ballistic movement of the foot with parameters.

Other research have obtained a walk in trot by a strait line with a velocity of  $0.1467 \text{ [m/s]}$ , for an AIBO [BHS00, p. 64]. This velocity was obtained with a walk where the knees were bent and the legs were moved in an angle to the body, as illustrated on Figure 4.12. Since the AIBO and NBT almost is the same size, should a velocity of  $0.1467 \text{ [m/s]}$  be obtainable for the NBT.

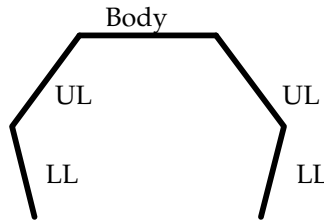


FIGURE 4.12: An illustration of the leg placement used in other research. The robot dog is illustrated from behind.

## 4.1 FALL ANALYSIS

There are two principals of balance there are of interest, these are; static- and dynamic balance. The principal of static balance is to keep the center of pressure inside the polygon formed by the feet supporting the weight of the body, an example of this with two supporting feet is illustrated on Figure 4.13, where the gray area is the polygon of support. The center of pressure is the projection of the NBT's CM on the floor.

The balance is not actively maintained in static balance, therefore must the velocity of the construction be kept low to secure that the center of pressure is kept inside the

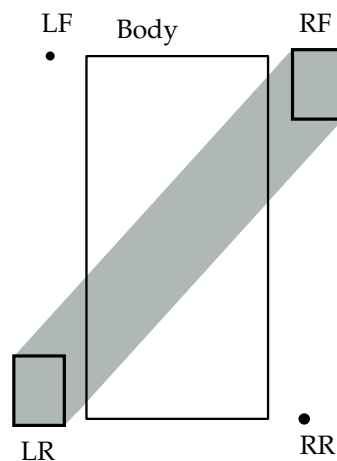


FIGURE 4.13: *An illustration of the standing area, seen from above. The gray polygon is the standing area with dynamic balance.*

polygon of support. The principal of dynamic balance require that the balance of the construction actively is maintained at all time by correcting the motions to avoid the falling, this requires a dynamically stable locomotion as trot is. The feet of the NBT is so small compared to the area they shall support, that they can be seen just as points, the support area then just will be a line. It is not possibly to keep the point of pressure on this line when walking, therefore dynamical balance is chosen.

The NBT will rotate around the line between the supporting feet, therefore can the balance be modeled as an inverted pendulum. The size of the rotations of the pendulum depends on the placement of the center of pressure, when the point is on the line is the pendulum not rotating if it is at rest, but if the point is beside the line is the gravity accelerating the CM against the ground. The force of gravity will rotate the pendulum faster the longer the center of pressure is from the line of support, therefore shall the the center of pressure be hold close to the line of support to hold the body steady doing trot. Therefore the design criteria for the movement of the legs will be to hold the center of pressure close to the line of support, the design depends of the manipulator kinematics model, therefore it will be designed with the path planner in Chapter 15.

When the NBT can move with a velocity of  $0.1467 \text{ [m/s]}$ , can it also be hit by another NBT with the same velocity. The maximum velocity of the NBT is unknown, therefore is the velocity  $0.1467 \text{ [m/s]}$  seen as the maximum velocity. The contact between two NBT's can be seen as an elastic collision between two particles, where particle one is the NBT getting hit strait from the side, and particle two is the NBT getting hit. The

worst case velocities after the collision are calculated in Equation (4.2).[SB00, p. 260-263]

$$\begin{aligned} v_{1f} &= \frac{m_1 - m_2}{m_1 + m_2} v_{1i} + \frac{2m_2}{m_1 + m_2} v_{2i} \\ &= v_{2i} \end{aligned} \quad (4.2a)$$

$$\begin{aligned} v_{2f} &= \frac{2m_1}{m_1 + m_2} v_{1i} + \frac{m_2 - m_1}{m_1 + m_2} v_{2i} \\ &= v_{1i} \\ &= 0[m/s] \end{aligned} \quad (4.2b)$$

The  $i$  mean before the collision and the  $f$  mean after the collision. It is two NBT there are colliding, therefore are the masses,  $m_1$  and  $m_2$  the same. The velocity  $v_{1i}$  is zero because the NBT not is moving sideways and the velocity  $v_{2i}$  is set to the velocity that is the goal to obtain for the NBT.

The force that the NBT get exposed to can be divided into two forces an example of these are illustrated on Figure 4.14. The line between the RF- and LR foot illustrates the axis that the body will turn around when it get hit.

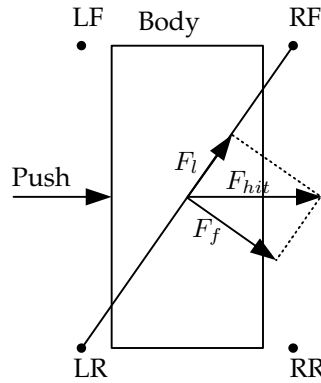


FIGURE 4.14: An illustration of how the force generated by a collision can be divided into other forces. The robot dog is illustrated from above, and the black dots are the placement of the feet.

$F_l$  is the force vector against the leg and  $F_f$  is the force vector making the NBT fall. The placement of the feet corresponding to the body gives the angle between the  $F_l$ ,  $F_f$  and  $F_{hit}$ , thereby controlling the size of  $F_l$  and  $F_f$ . The more the legs are rotated out to the side, the more  $F_f$  is going backwards. The slip resistant surface on the feet will have a better grip the more strait down on the floor the legs are, but this will decrease the standing area when all four feet are on the floor. The choice of an angle is a compromise between the balance, the achievable velocity of the locomotion and the slipping of the

feet. The compromise is chosen to be a rotation of 0.1 [rad] out from the body. This gives an angle of 0.9 [rad] between the leg and the floor, which will keep much of the pressure strait down on the floor, thereby preventing the feet from slipping and still giving a large standing area, which will give more stability when standing and shifting feet.

The model of the sensors will have to calculate the  $F_l$  and  $F_f$  to find the sensor output from the accelerometers. Before this can be done shall the angle of the line between the feet, that the body of the NBT will rotate around and the body be found. This angle is donated  $\psi_{IP}$  and is illustrated on Figure 4.15.

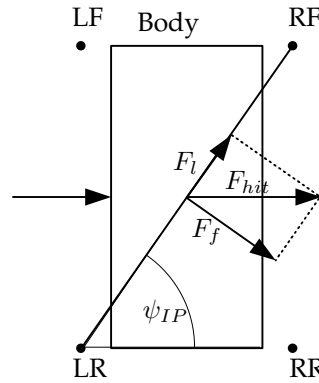


FIGURE 4.15: An illustration of how the angle between the body and the line between the feet supporting the body is calculated. The robot dog is illustrated from above, and the black dots are the placement of the feet.

A calculation for  $\psi_{IP}$  will not be derived here because it is strait forward to find when the kinematics model for the NBT is derived. The formulas for splitting  $F_{hit}$  into  $F_l$  and  $F_f$  are derived in equation (4.3).

$$F_l = \cos(\psi_{IP}) \cdot F_{hit} \quad (4.3a)$$

$$F_f = \sin(\psi_{IP}) \cdot F_{hit} \quad (4.3b)$$

The force against the supporting legs are from the motor in the shoulder of the RF seen as a torque, that will give an angle acceleration around the foot if there not is given a counter force from the motor. It is unknown how large the force of static friction for the slip resistant surface on the feet is, if  $F_l$  becomes larger then the force of static friction, the NBT will accelerate in the direction of  $F_l$ . The force of static friction is expected to be large enough to hold against  $F_l$  and since its size is unknown can't it be simulated, therefore the accelerating in the direction of the leg will be disregarded.

The force  $F_f$  will make a moment around the line between the supporting feet, this

is illustrated on Figure 4.16(a) and calculated in Equation (4.4).  $r_{CM}$  is the distance from the rotation point to the CM. The figure is seen along the line between the two standing feet. The points where the feet touch the ground make the rotation axis. The CM for the whole NBT is seen as the point  $P$ .  $\vartheta$  is the angle the point  $P$  is rotated around the standing feet. When the CM of the NBT not is strait over the standing feet will the gravity force make a moment which will make the NBT fall, this is illustrated on Figure 4.16.

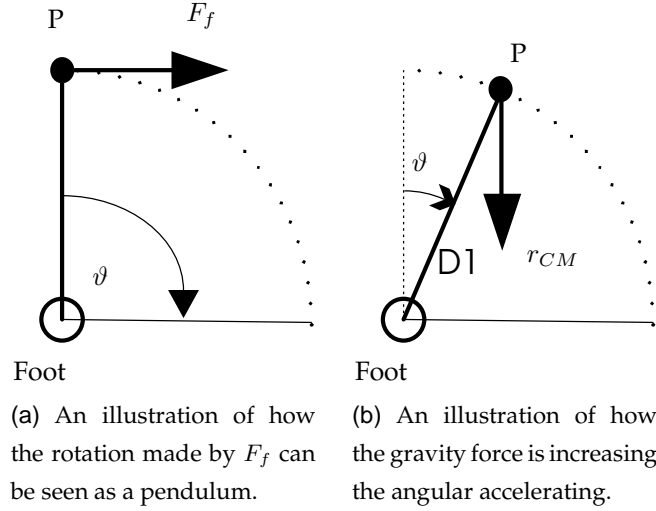


FIGURE 4.16: Illustrations of the angular accelerations.

$$\tau_f = r_{CM} F_f \cos(\vartheta) \quad (4.4)$$

$F_{g1}$  is the gravity force acting on the CM of the NBT, this is calculated in Equation (4.5a). The moment generated by the force  $F_{g1}$  is calculated in Equation (4.5b)

$$F_{g1} = m_{NBT} g \sin(\vartheta) \quad (4.5a)$$

$$\tau_{g1} = r_{CM} F_{g1} \sin(\vartheta) \quad (4.5b)$$

$$= r_{CM} m_{NBT} g \sin(\vartheta)$$

Small angles between the body and floor will occur when walking, these will be corrected doing walking, but if the angle gets to large can the fall of the inverted pendulum not be stopped doing walk, therefore has the walk to be stopped while the angle between the floor and body is corrected. The correction made by the balance to get the body parallel with the ground, depends on where in the walking sequence it is made. The correction can be made when the body is supported by two legs and the other two are moving for-

ward it can also be made when the body is supported by four legs. Both these scenarios, which will be described in the following:

**Supported by two legs and two are moving forward:** The walking is first stopped and then the two legs there were in the swing phase moved out from the side, in direction of the fall. The angle the legs are moved out to the side depends on the size of  $\tau_{g1}$ ,  $\tau_f$  and  $\tau_d$ . The total moment in the direction of the fall,  $\tau_{f,t}$  is calculated in Equation (4.6).

$$\begin{aligned}\tau_{f,t} &= \tau_f + \tau_{g1} + \tau_d \\ &= r_{CM} F_f \cos(\vartheta) + r_{CM} m_{NBT} g \sin^2(\vartheta) \\ &\quad + g \sin(\vartheta) t_d\end{aligned}\tag{4.6}$$

The two feet moved out to the side shall be placed in an angle to the floor so the gravity force produce a moment in the opposite direction of  $\tau_{f,t}$ , to stop the fall. This gravity force will be donated  $F_{g2}$ , and is illustrated on Figure 4.17.

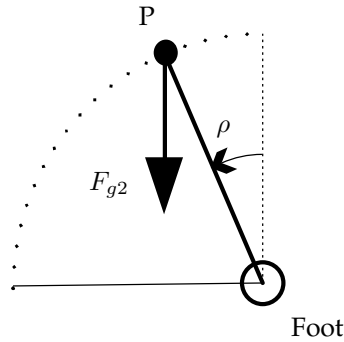


FIGURE 4.17: An illustration of how gravity in the opposite direction of  $F_{f,t}$  is made by placement of the foot.

$\rho$  is the angle from vertical to the line from the foot to the CM. The foot in the side that the NBT is falling toward has to stop the fall because the one in the other side can not get in an angle to produce a gravity force in the opposite direction of  $\tau_{f,t}$ . The angle  $\rho$  is calculated in Equation (4.7).

$$\begin{aligned}\tau_{g2} &= r_{CM} F_{g2} \sin(\rho) \\ &= r_{CM} m_{NBT} g \sin(\rho) \\ &= \tau_{f,t} \\ \rho &= \arcsin\left(\frac{r_{CM} m_{NBT} g}{\tau_{g2}}\right) \\ &= \arcsin\left(\frac{r_{CM} m_{NBT} g}{r_{CM} F_f \cos(\vartheta) + r_{CM} m_{NBT} g \sin^2(\vartheta) + g \sin(\vartheta) t_d}\right)\end{aligned}\tag{4.7}$$

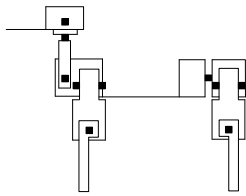
The angle is for the line from the CM to the placement of the foot in the side that the NBT is falling to. The leg there was in the swing phase in the NBT's opposite side shall be moved to the same angle if it is possibly. If the angle is outside the working area for the leg shall the leg be moved as much as possibly. The working ranges will be discussed further in the deriving of the inverse manipulator kinematics in Chapter 10.

**Supported by four legs:** If all four feet are on the ground, the swinging and standing legs are being shifted as a part of the walk cyclic. The walking cyclic is stopped if the angle between the body and floor is larger then a threshold. The two legs there should start a swing phase in the walking cyclic, are moved out as for when two legs are in the swing phase.

The considerations in this section will be the basis for deriving a model of the inverted pendulum.







# CENTRAL PATTERN GENERATOR

*This chapter analysis overall what a CPG is and how it can be used to update the model and generate walking trajectories.*

Research in biology suggests that all living creatures who have the ability to move one or more parts of their body, have a Central Pattern Generator (CPG). The CPG generates the rhythmic movements of the trunk and limb muscles, which together gives the movements of the body parts. These movements can be walk, swimming etc. The movements are over time corrected so the movements becomes better for the task. It is unknown how the structure of the CPG is adapted to improve the walking efficiency of an animal.

Researchers suggest that the model can be corrected by constructing an Adaptive Module (AM), which compares the actual physical movement with the expected movement. The physical movement is measured by sensors and the expected movement is calculated from a forward model which predicts the sensor measurements. The results of these are compared, if there are differences the model then will be updated by the CPG. This is illustrated on Figure 5.1. [LB02]

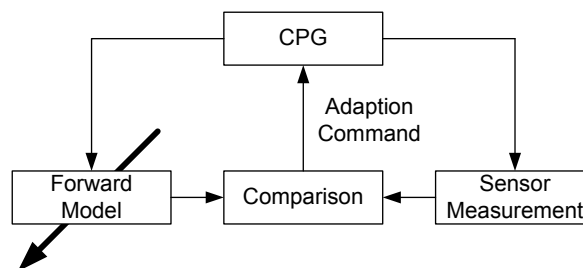
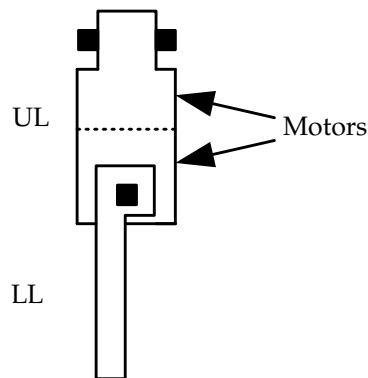


FIGURE 5.1: An illustration of how the AM could be constructed.

The updating is a complex task because the difference between the measured and expected result can be caused by differences in one or more of the body parts and disturbances from the environment. There can be differences in the angle the motors move to and the length between the joint and between the knee joint and the foot. The length of the UL is given by the length of two motors, as illustrated on Figure 5.2. The LL is a hand cut piece of aluminum. To find out witch parameter there have to be corrected, test cases have to be designed which gives unique solutions for which parameter there have to be corrected, and by haw much. This will be considered in the Chapter about CPG adaption.



**FIGURE 5.2:** *An illustration of the placements of the motors in the UL. The illustration is for a front leg, but the placement is similar in the rear legs.*

The CPG is also generating reference signals to the muscles for movements that the brain desires. Researchers suggest that the CPG can be seen as a state machine with different tasks. Other researchers have done this in different ways. One way has been to divide the CPG into unit CPG's which each control a small part of the robot [LB02]. The unit CPG then has a model for the part that it controls. The motions of the robot are coordinated by sending signals between the unit CPG's. Another way to generate reference signals has been to make one CPG, which generates the reference signals for all the joints [JFK95, p. 700-701]. This has earlier been done as a state machine with one state for each of the states described in Chapter 4. The coordination is the most important property in obtaining a stable locomotion, if the phase between the legs is out of synchronization, then the NBT will be falling [MZF04]. Both the principles can be used for generating trajectories for the walking and have been by other researchers, therefore the choice is difficult. The trajectory planner will be implemented as a state machine because the state machine is believed to be the easiest solution to implement and test because there is not a need to send signals between different parts to hold the phase between the legs. The suggested state machine is illustrated on Figure 5.3.

The quadruped robot is in the illustrations walking in the right direction. The dotted legs are the RF and LR, which are moved together. They will together be referred to as Leg Pair 1 (LP1) and the LF and RR will be referred to as Leg Pair 2 (LP2). The states B, C, D and E are together making the walk sequence illustrated on Figure 4.6 on page 20. The state is shifted when the desired positions for all the joints are obtained, thereby securing that the phase between the legs are synchronized. The positions for all the joints that the states shall obtain are called way points. These will be placed in the following. There

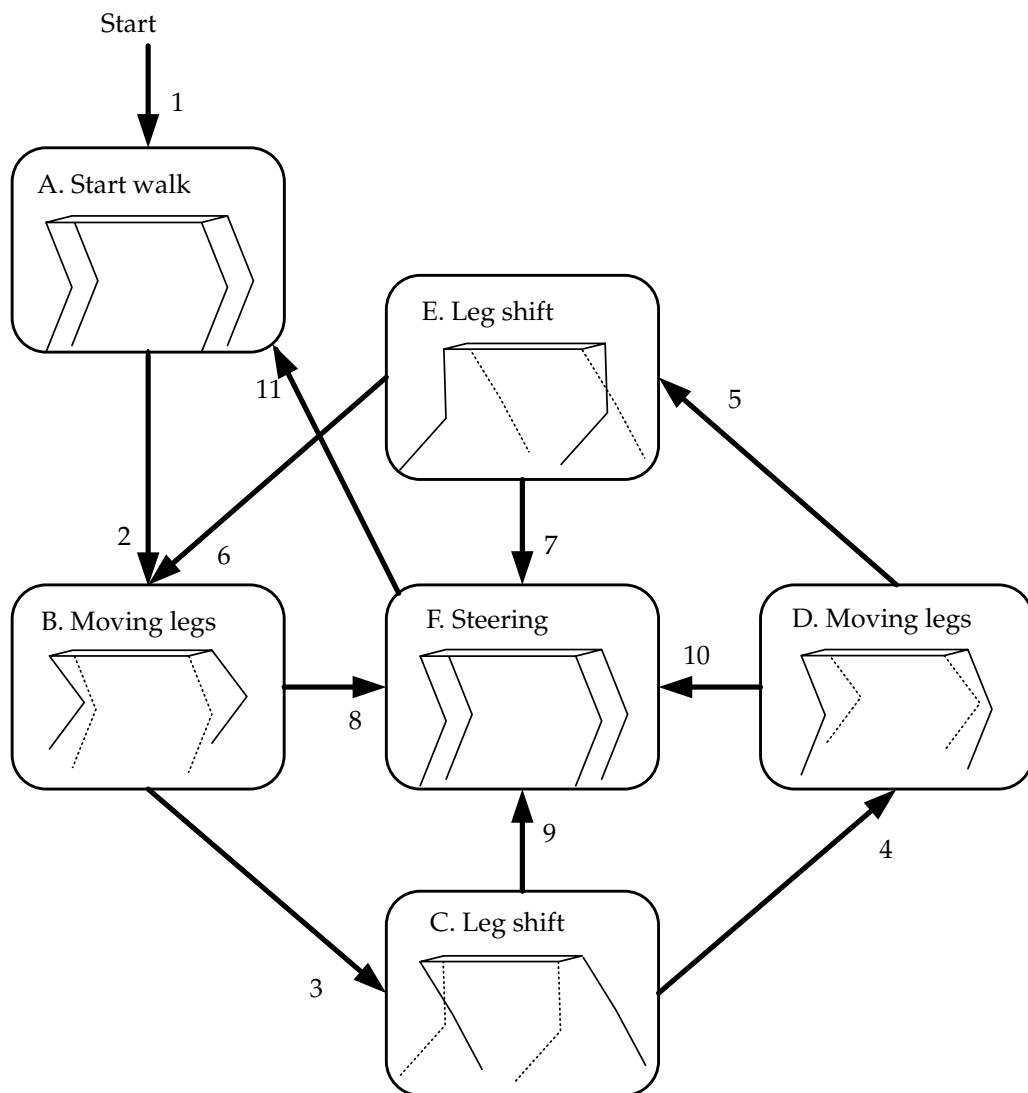


FIGURE 5.3: An illustration of how the state machine for the CPG could be constructed.

---

will be placed a way point when the foot is lifted and put down on the ground. It can be desirable to place more way points when the trajectory planning is being designed. The movement of the foot is stopped just when it is putted on the ground and raised from the ground, with the time for a swing or stand phase between. These are calculated in Equation (4.1).

The actions of the states on Figure 5.3 are described in the following:

**A. Start walk:** Initialize the walk sequence from a standing position. The feet of LP2 are lifted vertically up from the floor until they reaches the curve for the swing phase, while the legs in LP1 are supporting the weight of the NBT, with out moving. This is illustrated in Figure 5.4, where Figure 5.4(a) illustrates the standing and Figure 5.4(b) illustrates the raised and standing legs. The curves on Figure 5.4(b) are the trajectory for the foot described in Chapter 4.

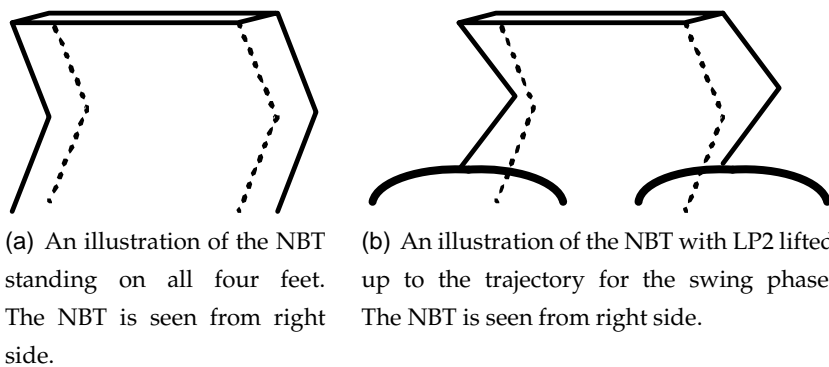


FIGURE 5.4: Illustrations of the start and end of phase A. start walk.

**B. Moving legs:** Are swinging LP2 forward and moving the body in over the feet of LP1. The movements are performed to a given time, which gives the NBT a certain velocity. In the start of the walk is the legs placed as Figure 5.4(b) illustrates. When the walk is started is the feet moved from the outer positions for the walk, these are defined in Chapter 4.

**C. Leg shift:** Are putting the feet of LP2 on the ground.

**D. Moving legs:** Are similar to B, just for the opposite Leg Pair (LP).

**E. Leg shift:** Are putting the feet of LP1 on the ground.

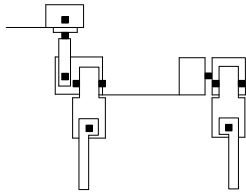
**F. Steering:** The steering control is always correcting to avoid falls, but this state is for when the angle of the body is to large to correct when walking. As described in Chapter 4 will the walk stop and the NBT will get the body parallel with the floor. Then will the legs be placed as illustrated on Figure 5.4(a).

The numbers marked on the arrows in Figure 5.3 indicates the conditions for the state shifts, these are listed in Table 5. All the conditions have to be fulfilled before the state can be shifted.

State shift	Description
1	When the walk sequence is started
2	When LP2 is lifted to the position illustrated on Figure 5.4(b) and LP1 is supporting the wait of the NBT as the figure illustrates.
3	When feet are moved to the end of their trajectory and is ready to be placed on the ground.
4	When all the feet are on the ground and the angle to the legs in LP1 is $\pi/4$ [rad] and $-\pi/4$ [rad] to the legs in LP2 as illustrated on Figure 15.2.
5	When feet are moved to the end of their trajectory and is ready to be placed on the ground.
6	When all the feet are on the ground and the angle to the legs in LP1 is $-\pi/4$ [rad] and the angle to the legs in LP2 is $\pi/4$ [rad] as illustrated on Figure 15.2.
7, 8, 9, 10	When the angle of the body is larger then the threshold described in Chapter 4 the pendulum has fallen so much that it not can be corrected doing walk.
11	When the body is parallel with the floor and the feet are in the position illustrated on Figure 5.4(a)

TABLE 5.1: Conditions for the state shifts in Figure 5.3





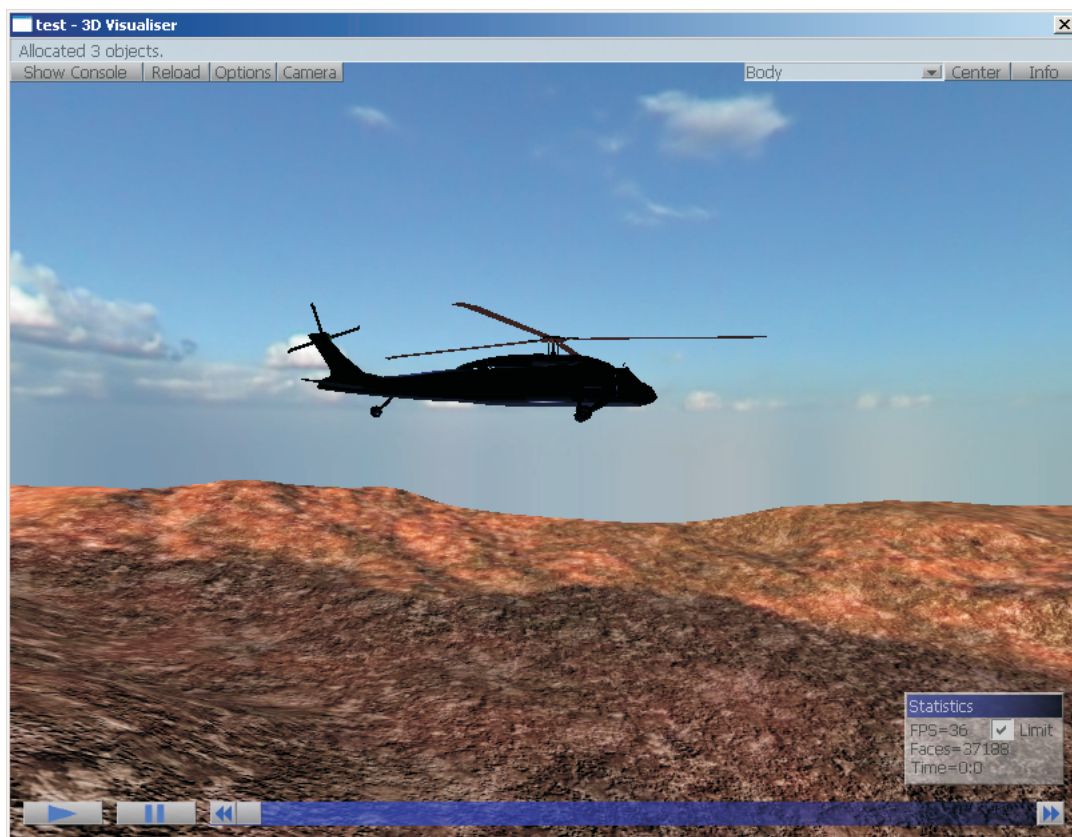
## VISUALIZATION

*This chapter analysis the visualization tool that will be used for the verifications, it also describes the changing of the field and the draws of the different parts of the NBT.*

The results of simulating the model and control will be difficult to make understand-ably only as graphs, therefore is it decided to visualize them in a 3D environment. For this GSim is chosen, which connect Simulink<sup>®</sup> to a 3D game engine. This connection is build on top of the TCP/IP protocol, which makes it possibly to use one computer for simulation and one for visualization. This is an advantage if the simulation is large and thereby uses much processor power or if the visualization consists of many objects and thereby used much processor power. [PG]

The communication is only one way, from Simulink<sup>®</sup> to the 3D engine. The data is send from Simulink<sup>®</sup> with the 3D Visualization Sink, which takes a nine dimensional vector as input, consisting of the position of the object, the angle of the object in degrease and the RGB color. The 3D Visualization Sink manages which object there shall be used in the visualization, the relation ship between the objects, the visualization time, etc.

When the 3D engine has received simulation results from Simulink<sup>®</sup>, the movement of the objects can be seen in a 3D visualization, as illustrated on Figure 6.1. The camera view can freely be changed in all three dimensions. The mountain landscape that is used by GSim has to be changed to a plain surface, because the walk is limited to this in Chapter 2 on page 7.



**FIGURE 6.1:** *Illustration of the landscape GSim has. The helicopter is a object rendered from Simulink<sup>®</sup> as the NBT will be.*



The ground surface is made by three maps; a *terrain\_heightmap.bmp*, a *terrain\_texture.jpg* and a *terrain\_detailmap.jpg*. The *terrain\_heightmap.bmp* is defining the curves of the landscape, this is altered to be flat. The *terrain\_texture.jpg* is defining the overall colors of the landscape and the *terrain\_detailmap.jpg* is defining the details to be inserted into the overall landscape. The landscape is designed with squares to give dept in the visualization and make it possibly to see the direction the objects are moving in. The landscape is demonstrated on Figure 6.2.

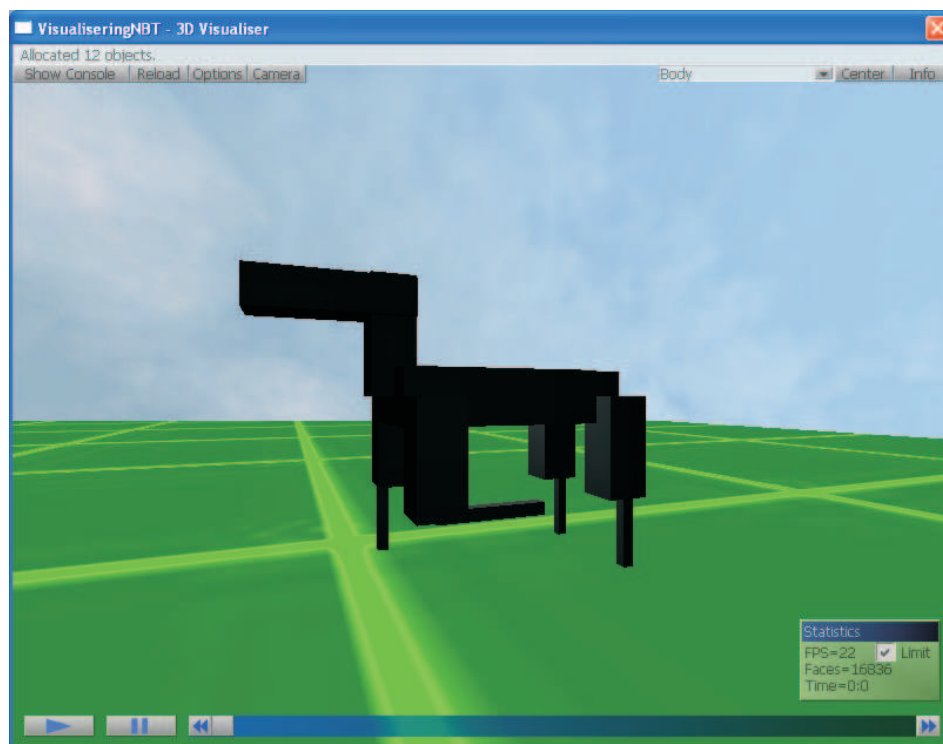
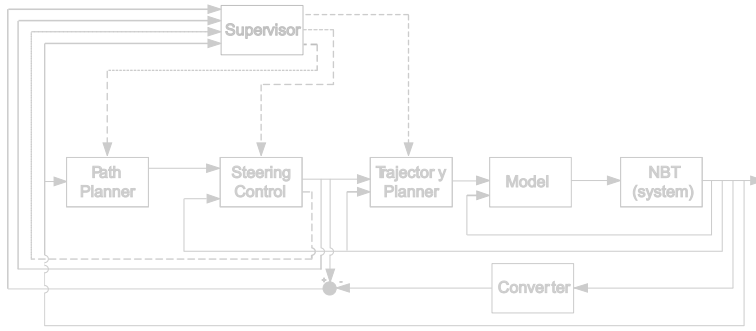


FIGURE 6.2: *Illustration of view in GSim after the field have been changed and the objects for the NBT have been inserted.*

The NBT will be drawn as objects of the type *3ds*, which will be made in the free ware drawing program *Anim8or*. The degree of detail of the objects will be limited so each object only consists of one 3D box, with the rotation axis placed in the joint axis. The drawing of the NBT is demonstrated on Figure 6.2.





# 7

## CONTROL SYSTEM ARCHITECTURE

*This chapter will provide a description of how the architecture of the control system will be and the input and outputs to each part of the project.*

The system will be combined so it fulfill the requirements stated in Chapter 2, and the things discovered in the analysis.

Only the controllers of high priority will be designed and implemented to limit the scope of the project. The *trajectory planner* and *path planner* will be designed and implemented because they makes the trajectories the feet shall follow and the path the NBT shall walk in, which the control use as input and reference.

The walk controller has by other researchers been implemented with success as P, PI and PID controllers [UoI] [HB00]. The NBT can be seen as an inverted pendulum when it is supported by two legs, a pendulum is unstable and will be falling. The *steering control* will correct the movements so the NBT can avoid falling when walking or getting in contact with other players. The CPG adaption will update the *model* for the given NBT for production differences, this will improve the movements and obtainable velocity of the walking. The walk controllers are very important when the system is implemented on the real NBT, but when simulating can the walk be performed with the control of the joints in open loop, therefore is the project limited from design and implementation of walk controller. The main goal of the project are to make the NBT walk and avoid falling, therefore will the focus be on the *steering control* and the CPG adaption will not be designed.

The decision maker described in Chapter 3 is choosing where to move to, this information is given to the *path planner* which is generating a path for the movement, therefore the *path planner* will be the first component in the system combining. The movement is in Chapter 2 limited to walking along a line, therefore the inputs to the *path planner* will be start walking and walk velocity, these signals will be denoted  $A_D$  and  $V_D$ , respectively. The values of  $A_D$  is 1 for start walk and else 0,  $V_D$  will be measured in meter pr. second. The output from the *path planner* shall define where the feet shall be moved to and to

---

which time. The position can be given in different ways, of interest are as; transformation matrices, position vectors or variables to shift between predefined trajectories. The predefined trajectories will minimize the online calculations on the NBT, thereby the use of processor power also will be minimized. The transformation matrices and position vectors will both have to give the positions of the feet relatively to a point fixed to the body, the transformation matrix will also give the information about the orientation of the LL. The *steering controller* is desirable to place before the *trajectory planner* because it then can correct the movements before the trajectory is calculated. The method with predefined trajectories shall have many trajectories to be able to shift smooth between them when the *steering controller* is avoiding a fall, therefore will this method not be used. When the position of the feet are known, the trajectory for the feet can be calculated, therefore the orientation of the LL will be unimportant because the position vector already have the needed information. Therefore the position vector will be chosen to define the desired position of the feet. The position vector will be denoted  $P_P$ , the period time  $t_{dP}$  and the time used in the period  $t_P$ .

Start walk  $A_D$  and desired walk velocity  $V_D$  is time discrete events that as earlier mentioned is controlled by the decision maker, which not is at disposal in this project, therefore will be made a *supervisor* controlling all the time discrete events. It will be designed as the state machine described in Chapter 5. The *supervisor* will also control where in the walk sequence the movements are, this will be denoted  $F_D$ , with the values; 1 for start walking, 2 for walking with LP2 in swing phase and 3 for walking with LP1 in swing phase.

The *steering control* will as earlier mentioned be placed between the *path planner* and *trajectory planner*. It will be designed to correct for the fall due to the unstableness of the inverted pendulum situation doing ordinary walk. If the angle of the body gets larger then a thresh hold  $\theta_D$  shall the *steering controller* stop the walk, get the body parallel with the floor again and ask the *supervisor* to start the walk again, this will be done by an output denoted  $A_B$ . The parameters of  $A_B$  will be 1 for stop walk, 2 for start walk and else 0.  $\theta_D$  is given by *supervisor* and is in radians around the longitudinal, latitudinal and vertically axis, these three axis are described in Chapter 3. The output position vectors from the *steering controller* will be denoted  $\bar{P}_B$ . The times  $t_P$  and  $t_{dP}$  can also be corrected by the *steering controller*, therefore will they be denoted  $t_B$  and  $t_{dB}$ , respectively.

The *trajectory planner* is giving a position vector for where the feet shall be to the given time, this position vector is denoted  $\bar{P}_T$ . To calculate these position vectors the *trajectory planner* shall know if the walk is being started or it is in the walk sequence, therefore will it get  $F_D$  from the *supervisor*.



---

joints, therefore it will consist of an *Inverse Manipulator Kinematics Model* and a *Manipulator Dynamics Model* as illustrated on Figure 7.2. Before these model parts can be derived a *Manipulator Kinematics Model* has to be derived, this will define the relationship between the different parts of the NBT.  $\bar{\theta}$  and  $\ddot{\theta}$  are shown as inputs, but they have to be calcu-

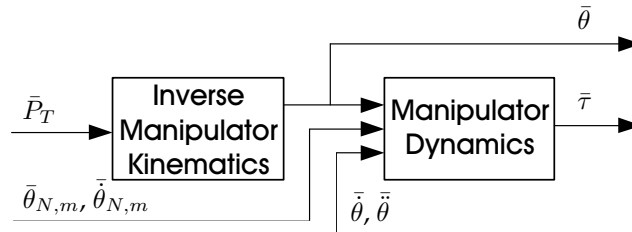
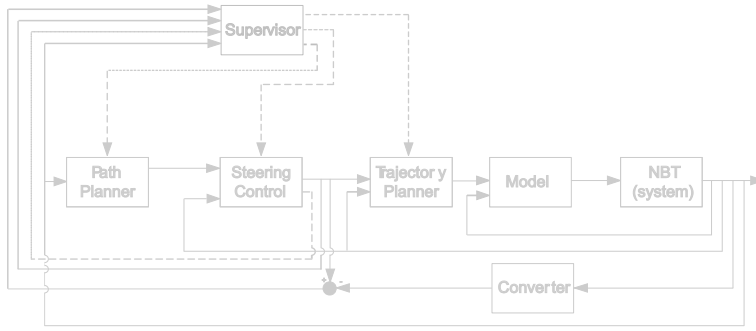


FIGURE 7.2: An illustration of the connections between the model parts.

lated either by converting the velocities and acceleration of the foot to the velocity and acceleration of the joints, this can be done by use of a Jacobian matrix. They can also be calculated by differentiating the calculated joint angle over the time. The method will be further discussed and one will be chosen doing the deriving of the *model*.



# 8

## CONTROL SYSTEM OBJECTIVES

The system shall be designed to fulfill the claims in Chapter 2 and in correlation with the system descriptions in Chapter 7. The primary goal of this project is to achieve autonomous walk along a straight line, with corrections to avoid falling. The walk shall be in trot, where the feet shall follow a ballistic looking curve. The walk shall be able to archive a velocity of  $0.1467 \text{ [m/s]}$ .

The steering control shall secure that the NBT not is falling on the ground. The system implementation will be limited from contacts with other players, but the influence of the contact shall be considered in the design.

Before the control can be designed a mathematical model have to be derived. This model will be derived as simple as possibly, because it is not the primary goal of this project.

The university of Newcastle already has a path planner, as described in Chapter 3, this are more sophisticated then the path planner in this project would be. Therefore will the path planner made in this project only be used for this project, therefore will it be made as simple as possibly. The path planner will be limited to start of walk and walking, this includes chancing of velocity, it will not include turning or stopping.

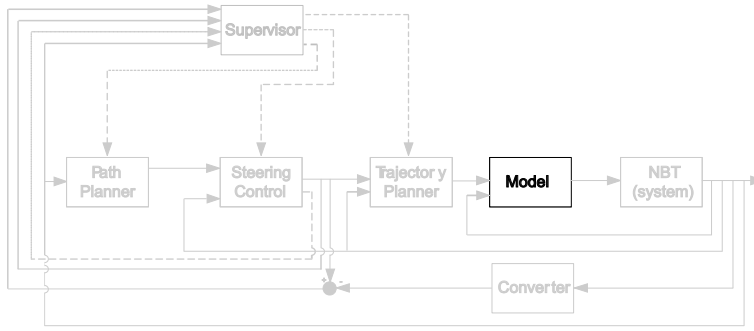
The trajectory planner shall make the trajectory for the feet both in the start walking and walking phase. The swing phase shall follow the parabolic curve derived in Chapter 3, where  $x$  is the movement along the floor and the plan spanned by the  $x$  and  $y$  is perpendicular to the floor. The start walk phase is more or less similar to the walk phase and the design will be redundant, therefore shall the design be as simple as possibly to conserve time for the other parts of the project.

The feedback signals shall be modeled, but since the real NBT is using sensors to measure the changes shall the sensor model be limited to be as simple as possibly.

The verification of whether the primary goal of the project is archived, shall be a 3D visualization of the NBT. This shall demonstrate the walk and the disturbances that the control system has to correct for. The model and control part also have to be verified individually, but this can also be done as is most convenient.





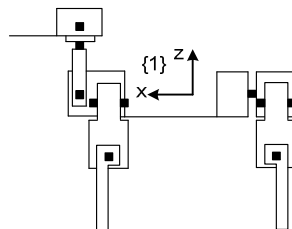


## MANIPULATOR KINEMATICS MODEL

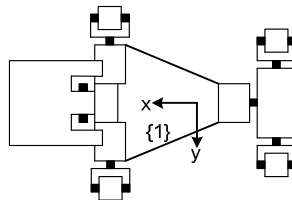
*This chapter concerns the derivation of the kinematics model of the NBT. The kinematics describes the geometrical relationship between the moving parts of the dog.*

As described in Chapter 7, the model shall give the rotational and linear, position, velocity and acceleration, for all the NBT's joints. To be able to calculate this, the geometrical relationship between the body, hip, UL, LL and feet has to be found.

First a body fixed reference coordinate system is placed on the NBT, for which the position of all the joints and the feet will be related to. The additional coordinate system is placed in the middle of the body with respect to the width and length, but in the same height as the first rotation axis of the shoulders. The placement and orientation of the coordinate system is illustrated in Figure 9.1.



(a) The NBT seen from the left side.



(b) The NBT seen from above.

**FIGURE 9.1:** Schematics of the NBT illustrating the placement of the additional coordinate system.

Figure 9.1(a) illustrates the NBT seen from its left side, and Figure 9.1(b) illustrates

---

the NBT seen from above. Both figures show the placement of the body fixed reference coordinate system, which is denoted  $\{1\}$ . This notation will be used throughout the report, for denoting coordinate systems. It is desirable to place an universal reference coordinate system to find the orientation, position, velocity and acceleration of the NBT in correlation to. Two different universal coordinate system are of interest; one fixed to the floor and one with the orientation and position that  $\{1\}$  should have. The one fixed to the floor is giving all the desired information, but the orientation in the plane parallel to the floor is not needed because there are limited from turning in Chapter 2 on page 7. The coordinate system following the NBT is desirable to kept parallel with the floor, because the rotation around the longitudinal- and latitudinal axis in the body can be calculated as the angle between the universal reference coordinate system and  $\{1\}$ . The universal coordinate system fixed to the floor is undesirable when the NBT is turned doing walk, because it then is walking in an angle to the universal reference coordinate system. The universal reference coordinate system with the orientation and position that  $\{1\}$  should have is chosen because the NBT always is walking along its x axis and the rotation around the longitudinal- and latitudinal axis is strait forward to calculate. The reference coordinate system is following the strait line with the same velocity and acceleration as the NBT has. The universal reference coordinate system is called  $\{0\}$ .

To be able to calculate the position, rotational- and linear-, velocity and acceleration for the joints, a coordinate system will be placed with respect to each joint. There will also be placed a coordinate system in each of the feet, to give the ability to calculate the position and orientation of the foot. There will be derived transformation matrixes to calculate between these coordinate systems. These transformation matrixes can be derived with different principles. The *Denavit-Hartenberg notation* will be used because it can derive the transformation matrix between neighboring joints in a robot arm and the legs of the NBT can be seen as four robot arms.

The *Denavit-Hartenberg notation* utilize four parameters [Cra05, chapter 3 p. 74-77]:

- $a_i$ : The distance from  $\hat{Z}_{i-1}$  to  $\hat{Z}_i$  measured along  $\hat{X}_{i-1}$
- $\alpha_i$ : The angle between  $\hat{Z}_{i-1}$  to  $\hat{Z}_i$  measured about  $\hat{X}_{i-1}$
- $d_i$ : The distance from  $\hat{X}_{i-1}$  to  $\hat{X}_i$  measured along  $\hat{Z}_{i-1}$
- $\theta_i$ : The angle between  $\hat{X}_{i-1}$  to  $\hat{X}_i$  measured along  $\hat{Z}_{i-1}$

The  $i$  refers to frame  $i$ , and  $i - 1$  refers to frame  $(i - 1)$ .  $\hat{X}_{i-1}$  and  $\hat{Z}_{i-1}$  are the unit vector for the  $X$  and  $Z$  axis in the  $\{i - 1\}$  coordinate system, respectively. The  $\hat{Z}_i$  axis shall point through the rotation axis of the joint.  $\theta_i$  is the angle of the joint, about  $\hat{Z}_{i-1}$  axis. The parameters  $\alpha_i$ ,  $a_i$  and  $d_i$  are used to place the coordinate system in a particular

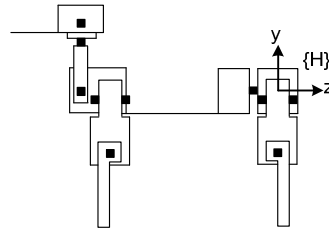
position, with a particular orientation.

If the frame is not placed in a joint, then it can be placed freely and  $\theta_i$  can be used to rotate the coordinate system to the  $\{i\}$  frame. It is important to realize that these rules only allow movement along, and rotation around, the  $X$  and  $Z$  axis. [Cra05, chapter 3 p. 74-77] The NBT has only revoluting joints and for the coordinate systems there are placed in joints, the rotation is given as  $\theta_i$ , with  $\alpha_i$ ,  $a_i$  and  $d_i$  constant. The values of  $\alpha_i$ ,  $a_i$  and  $d_i$  can be found in Appendix A on page 135. The four parameters are shown in equation (9.1), which gives the transformation matrix  ${}^{i-1}_iT(\theta_i)$ , from the  $\{i\}$  coordinate system to the  $\{i - 1\}$  coordinate system. [Cra05, p. 84] [Oga70, p. 140-143]

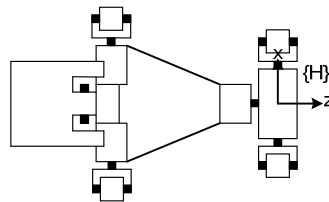
$${}^{i-1}_iT(\theta_i) = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) & 0 & a_i \\ \sin(\theta_i)\cos(\alpha_i) & \cos(\theta_i)\cos(\alpha_i) & -\sin(\alpha_i) & -\sin(\alpha_i)d_i \\ \sin(\theta_i)\sin(\alpha_i) & \cos(\theta_i)\sin(\alpha_i) & -\cos(\alpha_i) & -\cos(\alpha_i)d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (9.1)$$

A more detailed description of the *Denavit-Hartenberg notation* can be found in [Cra05, chapter 3 p. 74-77].

There is placed a coordinate system to make the rotations of the hip, allow that the back is stiff in trot locomotion. This is done because the model will be constructed so it can be used to other locomotion. Figure 9.2 illustrates the placement of the coordinate systems to make the rotations of the hip.



(a) The NBT seen from its left side.



(b) The NBT seen from above.

FIGURE 9.2: Schematics of the NBT illustrating the placements of coordinate system  $\{H\}$ .

The placement of  $\{H\}$  violates the rules in the *Denavit-Hartenberg notation* for trans-

formation between two neighboring coordinate systems. This can be solved by placing a coordinate system where both  $\{1\}$  and  $\{H\}$  can be transformed to. The two transformation matrices then will be multiplied together, this gives one transformation matrix between  $\{H\}$  and  $\{1\}$ . The transformation matrix can also be made from the unit vectors and a position vector. There are no difference between the result of the two solutions, since the *Denavit-Hartenberg notation* is preferable to used for the transformation matrices in the legs is this solution chosen. The placements of the extra coordinate systems will not be documented because it only is the ending transformation matrices there are interesting in this model. The same procedure is used in the feet to make the joint angle zero when the legs are in their initial position, which is straight.

Figure 9.3 and 9.4 illustrates the placement of the coordinate systems in the LF- and LR leg. The coordinate systems placed in the RF- and RR leg are placed as in the LF- and LR leg, respectively.

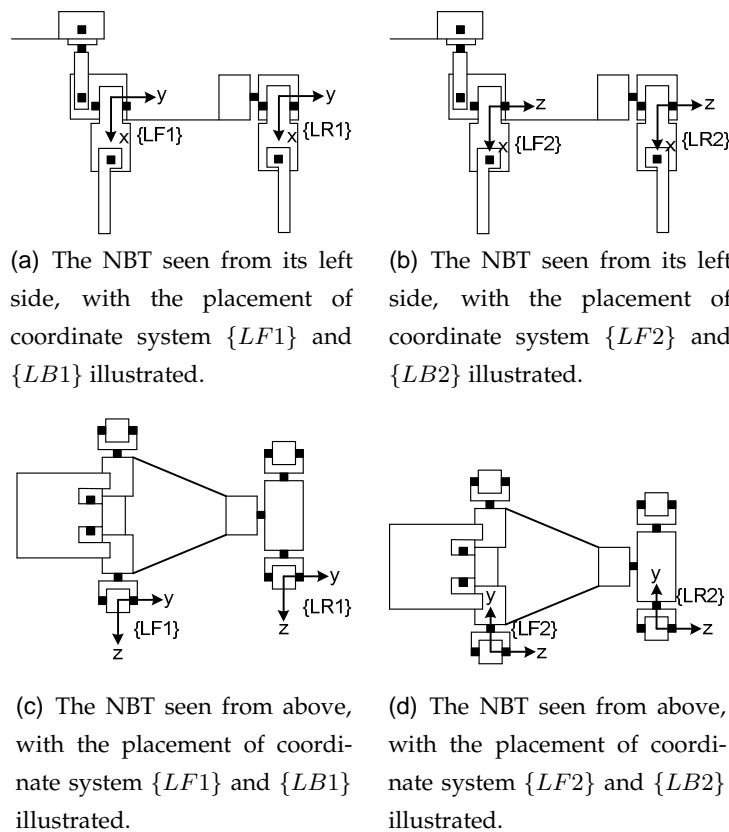


FIGURE 9.3: Schematics of the NBT illustrating the placements of the coordinate systems in the right front and back leg.

The relationship between  $\{1\}$  and the coordinate systems in the LF- and LR leg are illustrated on figure 9.5.

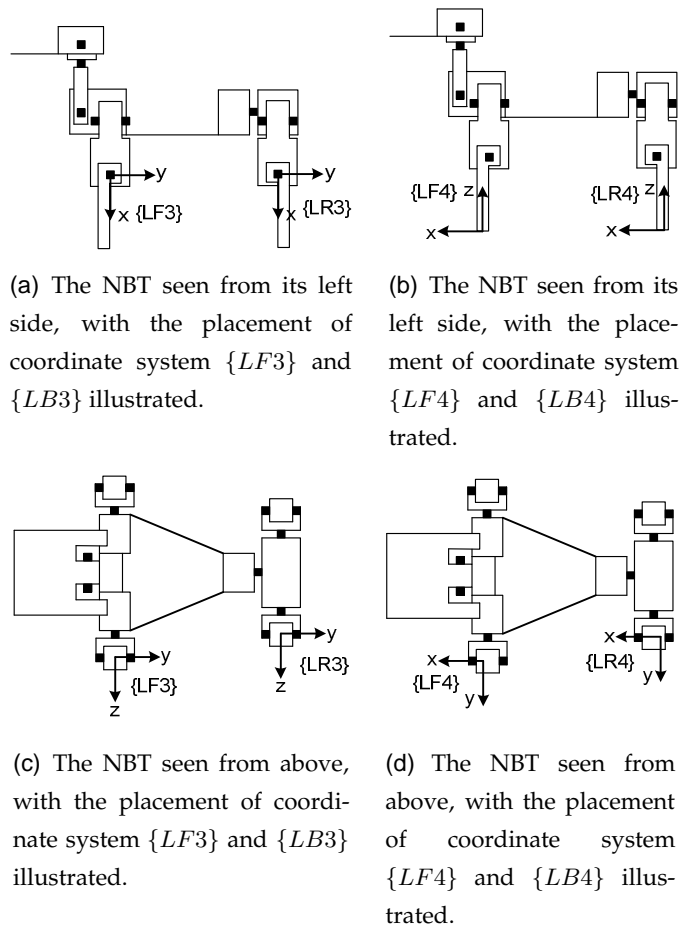


FIGURE 9.4: Schematics of the NBT illustrating the placements of the coordinate systems in the right front and back leg.

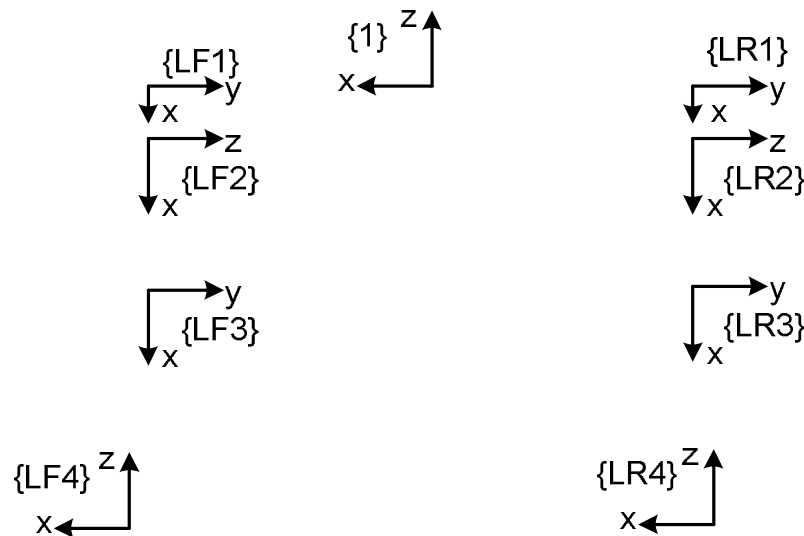


FIGURE 9.5: An illustration of the placement of the coordinate systems in the LF- and LR leg. The coordinate system  $\{1\}$  is also illustrated.

The calculation of the transformation matrixes are implemented in a script in MATLAB<sup>®</sup>. (Note: This can be found in `/simulation/model/kinematics.m`.) There are constructed functions to calculate the Coordinate Transformation Matrix (CTM) for each of the legs. (Note: These can be found in `/simulations/model/CTM_LF.m`, `/simulations/model/CTM_RF.m`, `/simulations/model/CTM_LR.m` and `/simulations/model/CTM_RR.m`, for the LF, RF, LR and RR legs, respectively.)

The correctness of the transformations are verified by giving the functions for the CTM different joint angles as input, where the transformation matrixes are known. The angle inputs are shown for the LF leg in Table 9.1. The calculated transformation matrixes are not documented in the report, but can be seen by running the verification in MATLAB<sup>®</sup>. (Note: The test can be found in `/simulations/model/KinematicstEST.m`.)

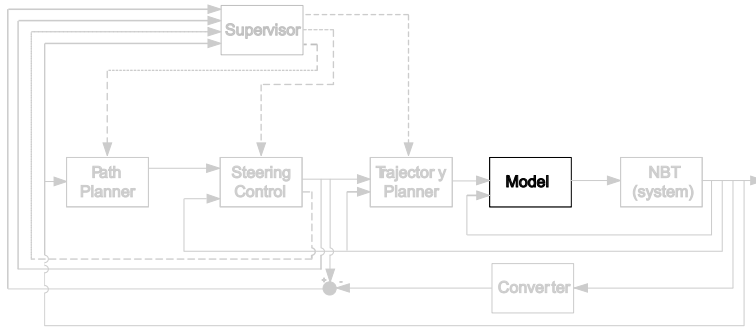
The verification of the calculation of the transformation matrixes gave the expected results.

	$\theta_{LF1}$	$\theta_{LF2}$	$\theta_{LF3}$
Test 1	0 [rad]	0 [rad]	0 [rad]
Test 2	$\pi/2$ [rad]	0 [rad]	0 [rad]
Test 3	$-\pi/2$ [rad]	0 [rad]	0 [rad]
Test 4	0 [rad]	$\pi/2$ [rad]	0 [rad]
Test 5	0 [rad]	$-\pi/2$ [rad]	0 [rad]
Test 6	0 [rad]	0 [rad]	$\pi/2$ [rad]
Test 7	0 [rad]	0 [rad]	$-\pi/2$ [rad]

**TABLE 9.1:** *The input values to verify that the transformation matrixes are calculated correct. The values are the same for the other legs.*

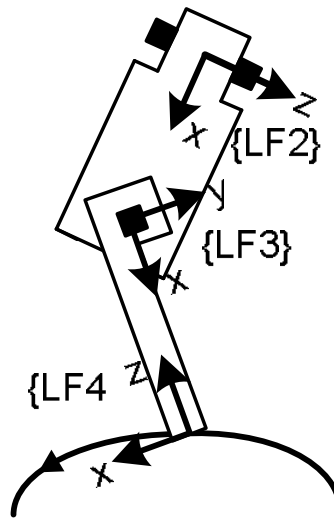






## INVERSE MANIPULATOR KINEMATICS MODEL

*This chapter concerns the derivation of the inverse manipulator kinematics model of the NBT. The inverse kinematics is a method to calculate the required joint angles to obtain the position and orientation of the foot for the given way point in the walking sequence. One placement of the leg is illustrated in Figure 10.1, where a method to calculate the joint angles shall be derived.*



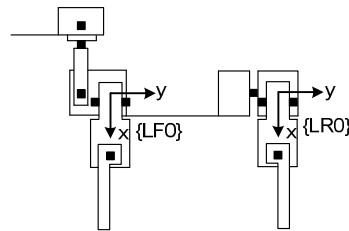
**FIGURE 10.1:** *Illustration of the movement of the NBT's leg in the swing phase.*

The kinematics designed in Chapter 9 will be used as a base for the derivation of the inverse manipulator kinematics. The transformation matrices and placements of coordinate systems can be found in Chapter 9.

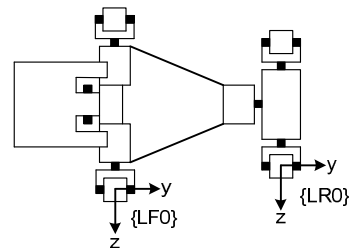
The method to find the joint angles can be based on different approaches, of interest are; algebraic and geometric approaches. The algebraic approach compares the kinematic relationship with the position vector for the desired position, from these relationships are formulas for the angles derived. The geometric approach is considering trigonometric relationships which can be used to calculate the joint angles. [Cra05, p. 109-128]

The algebraic approach is more computational demanding than the geometric approach. The geometric approach uses less computation effect, but is more difficult and takes more time to derive. Both the approaches will give more results for each joint which have to be compared to the position vector and the working ranges of the joints to find the correct joint angles. As mentioned in Chapter 3 has the NBT limited processor power, therefore is the geometrical approach chosen because it uses less computation effect.

The position vector for the location of the foot shall be given to a point fixed to the body of the NBT. To simplify the calculations and avoid calculating with offsets from the shoulder to the point fixed to the body, there will be placed an extra coordinate system in the shoulders, where the position vector for the foot location will be related to. The extra coordinate systems will be coinciding with  $\{LF1\}$ ,  $\{RF1\}$ ,  $\{LR1\}$  and  $\{RR1\}$  in their initial position and will be donated  $\{LF0\}$ ,  $\{RF0\}$ ,  $\{LR0\}$  and  $\{RR0\}$ , respectively. The coordinate systems  $\{LF0\}$  and  $\{LR0\}$  are illustrated on Figure 10.2.



(a) The NBT seen from its left side, with the placement of coordinate system  $\{LF0\}$  and  $\{LB0\}$  illustrated.



(b) The NBT seen from above, with the placement of coordinate system  $\{LF0\}$  and  $\{LB0\}$  illustrated.

**FIGURE 10.2:** Schematics of the NBT illustrating the placements of the coordinate systems in the right front and back leg.

The position vectors will be donated  ${}^{LF0}_{LF4}P_T$ ,  ${}^{RF0}_{RF4}P_T$ ,  ${}^{LR0}_{LR4}P_T$  and  ${}^{RR0}_{RR4}P_T$ , from the foot to the given coordinate system for the LF, RF, LR and RR, respectively.

The joint angle  $\theta_H$  for the hip will be set to zero because there is no rotations in the back when standing and walking in trot. The rotation for  $\theta_{LF1}$ ,  $\theta_{RF1}$ ,  $\theta_{LR1}$  and  $\theta_{RR1}$  in trot are unknown, therefore will the angles be limited to a maximum rotation of  $\pm\pi/2$  because the NBT then has the ability to lay down on the stomach, which are used by some teams in the Four-Legged League. The work space for the other angle in the shoulders,  $\theta_{LF2}$ ,  $\theta_{RF2}$ ,  $\theta_{LR2}$  and  $\theta_{RR2}$  will be limited to a maximum rotation of  $\pi/4$  in under the body and rotation of  $\pi/2$  away from the body. The rotation in under the body is limited to  $\pi/4$  because the leg will get in contact with the body at approximately this angle. The LL will be limited to a rotation between 0 and  $\pi/2$  because the LL only rotates backwards in trot.

The calculations of joint angles are akin for all the legs, therefore the deriving only will be documented for the LF leg.  $\theta_{LF2}$  is rotating a plan where the UL, LL,  $\theta_{LF3}$  and foot is moving in, this plan is rotated out from the plan that  $\theta_{LF1}$  is working in, if  $\theta_{LF2}$  has other value then the initial. The rotation is around the axis of  $\theta_{LF2}$ . When  $\theta_{LF2}$  is zero is the plans coinciding. The plan that  $\theta_{LF1}$  rotates in is donated *plan A* and the plan rotated by  $\theta_{LF2}$  is donated *plan B*, this is illustrated on Figure 10.3.

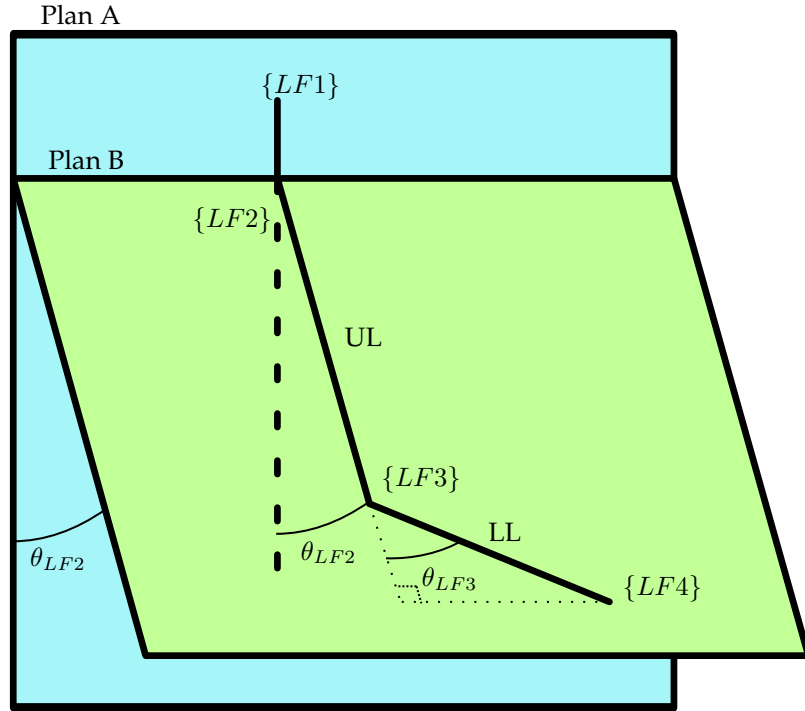


FIGURE 10.3: Illustration of how plan B is rotated in collation with plan A.

The formulas for calculating the joint angles are derived in Appendix B, the results of

this calculations can be seen in Equation (10.1).

$$\theta_{LF1} = \arctan \left( \frac{l_3 \sin(\theta_{LF3A})}{(l_2 + l_3 \cos(\theta_{LF3A})) \cos(\theta_{LF2}) + l_1} \right) - \arctan \left( \frac{P_{P,y}}{P_{P,x}} \right) \quad (10.1a)$$

$$\theta_{LF2} = -\arcsin \left( \frac{2 \cdot l_2 \cdot P_z}{(l_1^2 - l_2^2 + l_3^2 - L^2)^2 + 4 \cdot l_1^2 \cdot P_z^2} \right) - \arctan \left( \frac{2 \cdot l_1 \cdot P_z}{l_1^2 - l_2^2 + l_3^2 - L^2} \right) \quad (10.1b)$$

$$\theta_{LF3} = \begin{cases} \arccos \left( \frac{(l_1 + l_2)^2 + l_3^2 - L^2}{2(l_1 + l_2)l_3} \right) + \theta_{LF3,offset} & \text{for } \theta_{LF2} = 0 \\ \arccos \left( \frac{\frac{P_z}{\sin(\theta_{LF2})} - l_2}{l_3} \right) + \theta_{LF3,offset} & \text{else} \end{cases} \quad (10.1c)$$

$l_1$ ,  $l_2$  and  $l_3$  are equal to  $L_{LF1 \rightarrow LF2}$ ,  $L_{LF2 \rightarrow LF3}$  and  $L_{LF3 \rightarrow LF4}$ , respectively.  $P_x$ ,  $P_y$  and  $P_z$  are equal to  $\frac{LF0}{LF4}P_{T,x}$ ,  $\frac{LF0}{LF4}P_{T,y}$  and  $\frac{LF0}{LF4}P_{T,z}$ , respectively.  $\theta_{LF3A}$  is equal to  $\theta_{LF3}$  plus the offset angle made by the displacement in the knee, this is illustrated for a front leg on Figure 10.4, where  $\theta_{offset}$  is the offset angle.

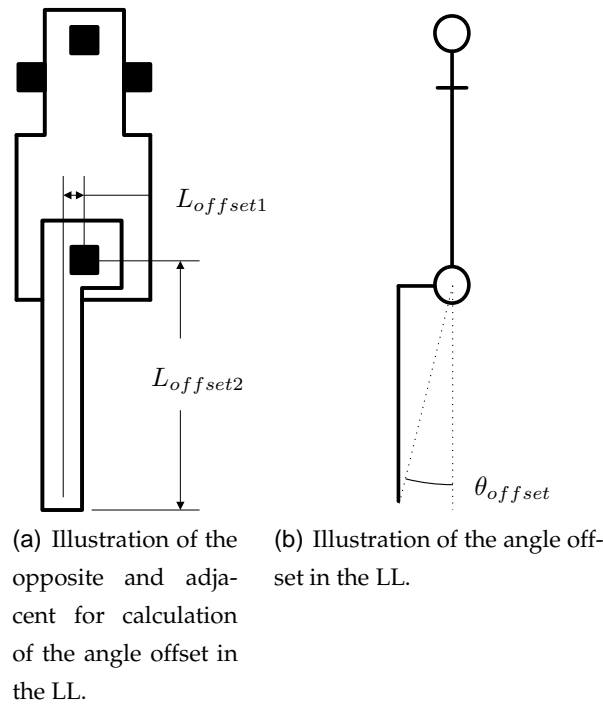


FIGURE 10.4: The illustration is viewed from the left side of the NBT.

The signs of the calculated angles shall be compared to the position vector and work space for the joint, and corrected if wrong.

The derived formulas for calculating the joint angles are implemented in a function. (Note: This can be found in `/simulations/model/IKinematics.m`.)

The implemented calculations to find the joint angles are tested, by giving the function different numerical position vectors, where the joint angles to obtain this position vector are known. There is constructed a function to calculate the numerical position vector from given joint angles, the function is using the manipulator kinematics derived in Chapter 9. (Note: The function can be found in `/simulations/model/position.m`) The test are similar for all the legs, therefore will only the test of the LF leg be documented, the tests for all the legs can be found in `/simulations/model/IkinematicstEST.m`. The given joint angles and the position vectors used in the tests of the LF leg can be found in Table 10.1. The results of the tests can be found in Table 10.2.

	$\theta_{LF1}$	$\theta_{LF2}$	$\theta_{LF3}$	Position vector
Test 1	0 [rad]	0 [rad]	0 [rad]	$[0.1632 \ -0.0110 \ 0.0000]^T$
Test 2	$\pi/2$ [rad]	0 [rad]	0 [rad]	$[0.0110 \ 0.1632 \ 0.0000]^T$
Test 3	$-\pi/2$ [rad]	0 [rad]	0 [rad]	$[-0.0110 \ -0.1632 \ 0.0000]^T$
Test 4	$\pi/4$ [rad]	0 [rad]	0 [rad]	$[0.1232 \ 0.1076 \ 0.0000]^T$
Test 5	$-\pi/4$ [rad]	0 [rad]	0 [rad]	$[0.1076 \ -0.1232 \ 0.0000]^T$
Test 6	0 [rad]	$\pi/2$ [rad]	0 [rad]	$[0.0110 \ -0.0110 \ 0.1522]^T$
Test 7	0 [rad]	$-\pi/2$ [rad]	0 [rad]	$[0.0110 \ -0.0110 \ -0.1522]^T$
Test 8	0 [rad]	$\pi/4$ [rad]	0 [rad]	$[0.1186 \ -0.0110 \ 0.1076]^T$
Test 9	0 [rad]	$-\pi/4$ [rad]	0 [rad]	$[0.1186 \ -0.0110 \ -0.1076]^T$
Test 10	0 [rad]	0 [rad]	$\pi/2$ [rad]	$[0.0876 \ 0.0866 \ 0.0000]^T$
Test 11	0 [rad]	0 [rad]	$\pi/4$ [rad]	$[0.1456 \ 0.0535 \ 0.0000]^T$

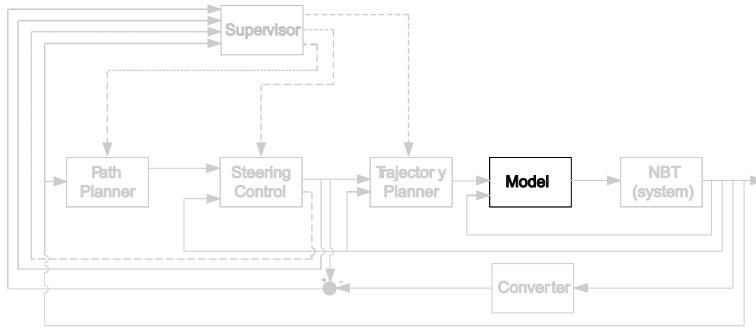
**TABLE 10.1:** *The input values to verify that the joint angles are calculated correct, with these joint angles is the position vectors calculated by use of the manipulator kinematics derived in Chapter 9. The joint angles used to test the other legs are the same.*

The angle input in test 6 is outside the working range for the left front leg, therefore is the result giving wrong values. There can be designed a algorithm to verify if the angles are correct and if they are wrong can they be correct on the fly or an error message can be send, but this is not the scope of this project, therefore will it not be discussed further. In the tests where the input to  $\theta_{LF3}$  were zero is it calculated to 0.0014 [rad] which is lower then the resolution of the actuators, therefore will there not be moved on this angle and it will then be the same as zero. The calculated angle for  $\theta_{LF1}$  in test 2 to 5 gave an offset smaller then  $\pm 0.07$  this is also accepted. The rest of the results gave the expected angle.

---

	$\theta_{LF1}$	$\theta_{LF2}$	$\theta_{LF3}$
Test 1	0 [rad]	0 [rad]	0.0014 [rad]
Test 2	1.5033 [rad]	0 [rad]	0.0014 [rad]
Test 3	-1.5033 [rad]	0 [rad]	0.0014 [rad]
Test 4	0.7155 [rad]	0 [rad]	0.0014 [rad]
Test 5	-0.8507 [rad]	0 [rad]	0.0014 [rad]
Test 6	-1.4783 [rad]	0.7854 [rad]	0.1277 [rad]
Test 7	0 [rad]	-1.5708 [rad]	0.0014 [rad]
Test 8	0 [rad]	0.7854 [rad]	0.0014 [rad]
Test 9	0 [rad]	-0.7901 [rad]	0 [rad]
Test 10	0 [rad]	0 [rad]	1.5722 [rad]
Test 11	0 [rad]	0 [rad]	0.7868 [rad]

**TABLE 10.2:** *The calculated joint angles from the test inputs in Table 10.1.*

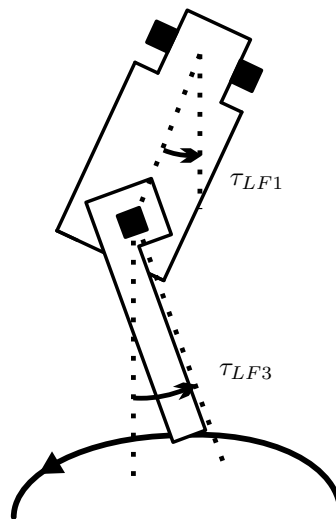


11

# MANIPULATOR DYNAMICS

## MODEL

*This chapter concerns the derivation of the dynamics of the NBT. The dynamic model calculates the torque required to move to and hold the joint in a given position, this is illustrated on Figure 11.1.*



**FIGURE 11.1:** *Illustration of two of the torques for the LF leg. All torques in the joints for all the NBT's four legs will be derived method to calculate in this chapter. The foot is following the parabolic curve that were described in Chapter 4 on page 17.*

The control system architecture state that there shall be developed an automatic method to calculate the torque required to hold and move the joints to given positions. To move a link to a given position it must be accelerated and then decelerated again. The torque required to accomplish this is a function of the desired acceleration and the mass distribution of the link. The mass distribution is a combination of the CM and Inertia Tensor (IT) of the link. [Cra05, p. 190-195]

The automatic method to calculate the torques can be derived with different meth-

ods, of interest are Newton-Euler dynamic formulation and Lagrangian mechanics. The Newton-Euler dynamic formulation is a force consideration, this is used in [BG96, p. 419-428]. The Lagrangian mechanics is an energy consideration, this is used in [TwY99, p. 59-70]. The dynamic method to calculate the torque will be implemented with the Newton-Euler dynamic formulation because this is the most efficient method from the view point of computational efficiency [TwY99, p. 59-60]. This is important because the NBT has limited processor power, as mentioned in chapter 3.

The Iterative Newton-Euler dynamic formulation consists of two parts, outwards- and inwards iterations [Cra05, chapter 6 p. 196-200]. The formulas for the outwards iterations are shown in Equation (11.1), where  $i$  is the current joint and  $i + 1$  is the next joint.

$${}^{i+1}\omega_{i+1} = {}^i{}^{i+1}R {}^i\omega_i + \dot{\theta}_{i+1} {}^{i+1}\hat{Z}_{i+1} \quad (11.1a)$$

$${}^{i+1}\dot{\omega}_{i+1} = {}^i{}^{i+1}R {}^i\dot{\omega}_i + {}^i{}^{i+1}R {}^i\omega_i \times \dot{\theta}_{i+1} {}^{i+1}\hat{Z}_{i+1} + \ddot{\theta}_{i+1} {}^{i+1}\hat{Z}_{i+1} \quad (11.1b)$$

$${}^{i+1}\dot{v}_{i+1} = {}^i{}^{i+1}R ({}^i\dot{\omega}_i \times {}^iP_{i+1} + {}^i\omega_i \times ({}^i\omega_i \times {}^iP_{i+1}) + {}^i\dot{v}_i) \quad (11.1c)$$

$${}^{i+1}\dot{v}_{cm_{i+1}} = {}^{i+1}\dot{\omega}_{i+1} \times {}^{i+1}P_{cm_{i+1}} + {}^{i+1}\omega_{i+1} \times ({}^{i+1}\omega_{i+1} \times {}^{i+1}P_{cm_{i+1}}) \quad (11.1d)$$

$$+ {}^i\dot{v}_i \quad (11.1e)$$

$${}^{i+1}F_{i+1} = m_{i+1} {}^{i+1}\dot{v}_{cm_{i+1}} \quad (11.1f)$$

$${}^{i+1}N_{i+1} = {}^{c_{i+1}}I_{i+1} {}^{i+1}\dot{\omega}_{i+1} + {}^{i+1}\omega_{i+1} \times {}^{c_{i+1}}I_{i+1} {}^{i+1}\omega_{i+1} \quad (11.1g)$$

$\dot{\theta}_{i+1}$  is the rotational velocity,  $\ddot{\theta}_{i+1}$  is the angular acceleration, for link  $i+1$ .  ${}^{i+1}\dot{v}_{i+1}$  is the linear acceleration of frame  $i + 1$  in origo of  $i + 1$  and  ${}^{i+1}\dot{v}_{cm_{i+1}}$  is the linear acceleration of frame  $i + 1$  in CM of  $i + 1$ .  ${}^{i+1}F_{i+1}$  is the force acting on frame  $i + 1$  and  ${}^{i+1}N_{i+1}$  is the moment acting on frame  $i + 1$ .  ${}^{i+1}\omega_{i+1}$  is the rotational velocity given from the former joints, and  ${}^{i+1}\dot{\omega}_{i+1}$  is the angular acceleration given by the angular acceleration of the former joints, these are for the first joints in the legs, the angular velocity and acceleration for the body, respectively.  ${}^i{}^{i+1}R$  is the rotation matrix from  $i$  to  $i + 1$  and  ${}^iP_{i+1}$  is the position vector from origo of  $i$  to origo of  $i + 1$ .  ${}^{i+1}P_{cm_{i+1}}$  is the position vector to the CM of the  $i + 1$  frame from the origo of  $i + 1$ .  ${}^{i+1}\hat{Z}_{i+1}$  is this vector  $[0 \ 0 \ 1]^T$ .  $m_{i+1}$  is the mass of the  $i + 1$  element.  ${}^{c_{i+1}}I_{i+1}$  is the inertia tensor of element  $i + 1$ , this is briefly described in Appendix C on page 143 [Cra05, section 6.3 p. 190-195].

The outwards iterations for the LF leg starts from  $\{1\}$  to  $\{LF1\}$  where the force and torque acting on the link is calculated. Following, the force and torque for  $\{LF1\}$  to  $\{LF2\}$  and  $\{LF2\}$  to  $\{LF3\}$  are calculated.



The formulas for the inwards iterations are shown in Equation (11.2).

$${}^i f_i = {}^i_{i+1} R^{i+1} f_{i+1} + {}^i F_i \quad (11.2a)$$

$${}^i n_i = {}^i N_i + {}^i_{i+1} R^{i+1} n_{i+1} + {}^i P_{C_i} \times {}^i F_i + {}^i P_{i+1} \times {}^i_{i+1} R^{i+1} f_{i+1} \quad (11.2b)$$

$$\tau_i = {}^i n_i^T {}^i \hat{Z}_i \quad (11.2c)$$

${}^i f_i$  is the force extended on link  $i$  by link  $i - 1$  and  ${}^i n_i$  is the torque extended on link  $i$  by link  $i - 1$ .  $\tau_i$  is the required joint torque to accelerate, decelerate or hold the position of joint  $i$ .

The inward iterations are calculated first from  $\{LF4\}$  to  $\{LF3\}$ , then from  $\{LF3\}$  to  $\{LF2\}$  and finally from  $\{LF2\}$  to  $\{LF1\}$ . The algorithm to calculate the inwards and outwards iterations is described in [Cra05, p. 200].

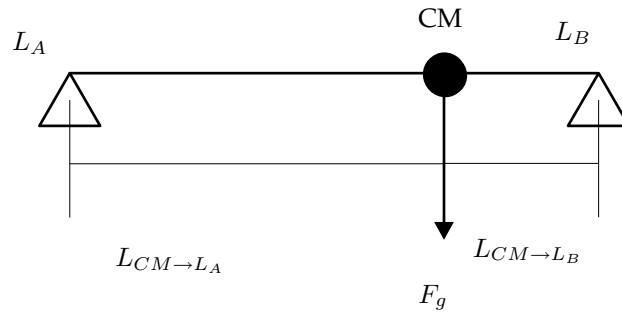
The acceleration due to gravity can be implemented in the algorithm by giving the shoulder an upwards acceleration equal to the downwards acceleration of the entire leg, as the outwards and inwards iterations give the same effect as implementing the gravity in each link. The gravity in the shoulders are giving a resulting normal forces from the floor in the opposite direction. [Cra05, p. 200]

The gravity force influence on each leg depends on the location of the CM for the entire NBT. This CM is moving when the legs are moving, thereby changing the gravity in the shoulders. Therefore is the location of the CM influencing on the torque needed in the joints. In construction of buildings and bridges it is important to calculate the force on the support points to secure that they are strong enough to hold the construction. The calculation of the force on the supporting points can be used for calculating the gravity force in the shoulders. The calculation of the force in the shoulders depend on the number of supporting legs, it can have two or four supporting legs. When the NBT has two legs on the ground the gravity force,  $F_g$ , is divided between them. The location of the CM gives the ratio for the splitting of  $F_g$ , these are illustrated on Figure 11.2.

$L_A$  and  $L_B$  are the supporting legs and  $L_{CM \rightarrow L_A}$  and  $L_{CM \rightarrow L_B}$  are the lengths from the CM to  $L_A$  and  $L_B$ , respectively. The distances from the CM to the shoulders shall be measured parallel to the floor in the calculations of the force in the shoulders. The gravity force in the shoulders are calculated in Equation (11.3).

$$F_{g,A} = \frac{L_{CM \rightarrow L_B}}{L_{CM \rightarrow L_A} + L_{CM \rightarrow L_B}} F_g \quad (11.3a)$$

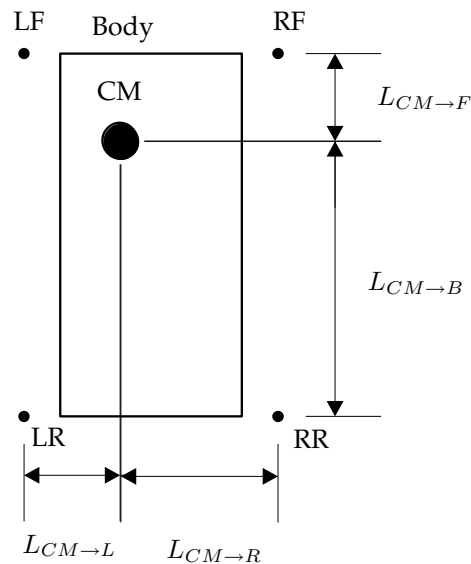
$$F_{g,B} = \frac{L_{CM \rightarrow L_A}}{L_{CM \rightarrow L_A} + L_{CM \rightarrow L_B}} F_g \quad (11.3b)$$



**FIGURE 11.2:** Illustration of the point load diagram with two supporting legs, with the lengths to the legs illustrated.

$F_{g,A}$  and  $F_{g,B}$  are the gravity force in  $L_A$  and  $L_B$ , respectively. If the CM not is on the line between the supporting legs will the gravity start an angle acceleration around the feet of the supporting legs, as described in Chapter 4 on page 17.

The calculations of the gravity force in the shoulders when supported by four legs is akin to the calculations when supported by two legs,  $F_g$  is just divided one more time. The lengths from the CM are illustrated on Figure 11.3.



**FIGURE 11.3:** Illustration of the lengths from the CM to the lines between two neighbouring legs.

$L_{CM \rightarrow L}$ ,  $L_{CM \rightarrow R}$ ,  $L_{CM \rightarrow F}$  and  $L_{CM \rightarrow B}$  are the length from the CM to left-, right side, front and back respectively. The gravity force in the shoulders are calculated in Equa-

tion (11.4).

$$F_{g,LF} = \frac{L_{CM \rightarrow R}}{L_{CM \rightarrow L} + L_{CM \rightarrow R}} \frac{L_{CM \rightarrow R}}{L_{CM \rightarrow F} + L_{CM \rightarrow B}} F_g \quad (11.4a)$$

$$F_{g,RF} = \frac{L_{CM \rightarrow L}}{L_{CM \rightarrow L} + L_{CM \rightarrow R}} \frac{L_{CM \rightarrow R}}{L_{CM \rightarrow F} + L_{CM \rightarrow B}} F_g \quad (11.4b)$$

$$F_{g,LR} = \frac{L_{CM \rightarrow R}}{L_{CM \rightarrow L} + L_{CM \rightarrow R}} \frac{L_{CM \rightarrow F}}{L_{CM \rightarrow F} + L_{CM \rightarrow B}} F_g \quad (11.4c)$$

$$F_{g,RR} = \frac{L_{CM \rightarrow L}}{L_{CM \rightarrow L} + L_{CM \rightarrow R}} \frac{L_{CM \rightarrow F}}{L_{CM \rightarrow F} + L_{CM \rightarrow B}} F_g \quad (11.4d)$$

$F_{g,LF}$ ,  $F_{g,RF}$ ,  $F_{g,LR}$  and  $F_{g,RR}$  are the gravity force in the LF-, RF-, LR- and RR leg, respectively.

A more detailed description of the dividing of the gravity force can be found in Appendix D on page 145. (Note: The dividing of  $F_g$  is implemented in a function, which can be found in `/simulations/model/GravityDeviding.m`. The calculation of the CM can be found in `/simulations/model/calcm.m`.)

The resulting normal forces from the floor are converted to be given in  $\{LF4\}$ ,  $\{RF4\}$ ,  $\{LR4\}$  and  $\{RR4\}$  for the given leg.

The torques for the joints will be represented as a state-space equations for each leg, these can be found in Equation (11.5). [Cra05, p. 190-191]

$$\bar{\tau}_{LF} = \bar{M}_{LF}(\bar{\theta}_{LF}) \bar{\ddot{\theta}}_{LF} + \bar{V}_{LF}(\bar{\theta}_{LF} \bar{\dot{\theta}}_{LF}) + \bar{G}_{LF}(\bar{\theta}_{LF}) \quad (11.5a)$$

$$\bar{\tau}_{RF} = \bar{M}_{RF}(\bar{\theta}_{RF}) \bar{\ddot{\theta}}_{RF} + \bar{V}_{RF}(\bar{\theta}_{RF} \bar{\dot{\theta}}_{RF}) + \bar{G}_{RF}(\bar{\theta}_{RF}) \quad (11.5b)$$

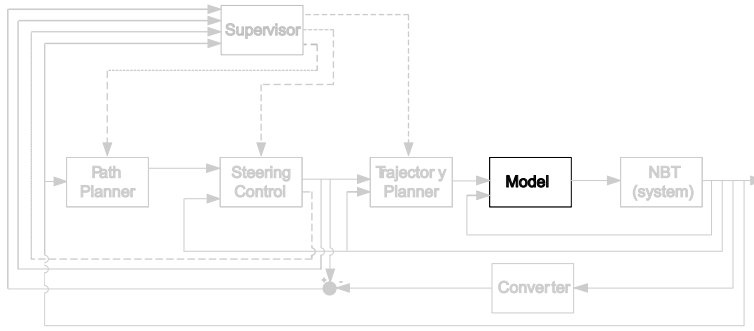
$$\bar{\tau}_{LR} = \bar{M}_{LR}(\bar{\theta}_{LR}) \bar{\ddot{\theta}}_{LR} + \bar{V}_{LR}(\bar{\theta}_{LR} \bar{\dot{\theta}}_{LR}) + \bar{G}_{LR}(\bar{\theta}_{LR}) \quad (11.5c)$$

$$\bar{\tau}_{RR} = \bar{M}_{RR}(\bar{\theta}_{RR}) \bar{\ddot{\theta}}_{RR} + \bar{V}_{RR}(\bar{\theta}_{RR} \bar{\dot{\theta}}_{RR}) + \bar{G}_{RR}(\bar{\theta}_{RR}) \quad (11.5d)$$

Where the  $\bar{M}(\bar{\theta})$  are the  $3 \times 3$  mass matrices for the manipulators,  $\bar{V}(\bar{\theta} \bar{\dot{\theta}})$  are  $3 \times 1$  vectors of the coriolis and centrifugal terms,  $\bar{G}(\bar{\theta})$  are  $3 \times 1$  vectors of the gravity.  $\bar{\tau}$ ,  $\bar{\theta}$ ,  $\bar{\dot{\theta}}$  and  $\bar{\ddot{\theta}}$  are  $3 \times 1$  vectors for the torques, joint angles, angular velocity and angular acceleration, respectively. The value of all the variables in Equation (11.5) can be calculated in `/simulations/model/dynamics_formula.m` and a simplified version can be found in Appendix D on page 145. The torque calculations are implemented as a function for each leg. (Note: The calculation of the torque for the joints can be found in `/simulations/model/dynamicsLF.m`, `/simulations/model/dynamicsRF.m`, `/simulations/model/dynamicsLR.m` and `/simulations/model/dynamicsRR.m`.)

---

The functions are structural tested, and the calculated joint torques were as expected. The input values for this test can be found in appendix D.3 on page 150.



## STRUCTURAL MODEL TEST

*This chapter concerns the test of the model. The testing will be based on a structural test, investigating the overall correctness of the model and the signs used.*

The tests should verify the calculation of the joint angles to obtain a given position and orientation of the foot, and also the torque required, this includes the torque required to begin moving, stop moving and holding the joint in a given position.

Before the tests can be conducted, the model parts have to be combined to one. The model will be combined to a model for each leg, this procedure are similar for all the legs, therefore the combining of the model for the LF leg only will be documented. The model parts are combined in Simulink<sup>®</sup>, therefore the diagrams in this chapter are produced from Simulink<sup>®</sup>. For a description of the standard Simulink<sup>®</sup> components, see the producers homepage at [www.mathworks.com](http://www.mathworks.com). The different parts of the diagrams are referred to by the name beneath them.

The LF leg is combined in a subsystem, this is illustrated in Figure 12.1. *Calculation of angles* and *Calculation of torques and forces* are Embedded MATLAB<sup>®</sup> functions which directly call the functions made in Chapter 10 on page 55 and Chapter 11 on page 61 for the LF leg, respectively.

The three blocks called *Calculation of  $\theta_{dot\_LF1}$  and  $\theta_{2dot\_LF1}$* , *Calculation of  $\theta_{dot\_LF2}$  and  $\theta_{2dot\_LF2}$*  and *Calculation of  $\theta_{dot\_LF3}$  and  $\theta_{2dot\_LF3}$*  are subsystems calculating the velocity ( $\dot{\theta}$ ) and acceleration ( $\ddot{\theta}$ ) for  $\{LF1\}$ ,  $\{LF2\}$  and  $\{LF3\}$ , respectively.  $\dot{\theta}$  and  $\ddot{\theta}$  can be found by use of Jacobian formalism or differentiate  $\theta$ , both methods can be used, therefore is differentiation chosen because it is fastest to implement and this is not the scope of the project. The blocks are alike, therefore only *Calculation of  $\theta_{dot\_LF2}$  and  $\theta_{2dot\_LF2}$*  will be described. Figure 12.2 on the following page illustrates the contents of the subsystem.

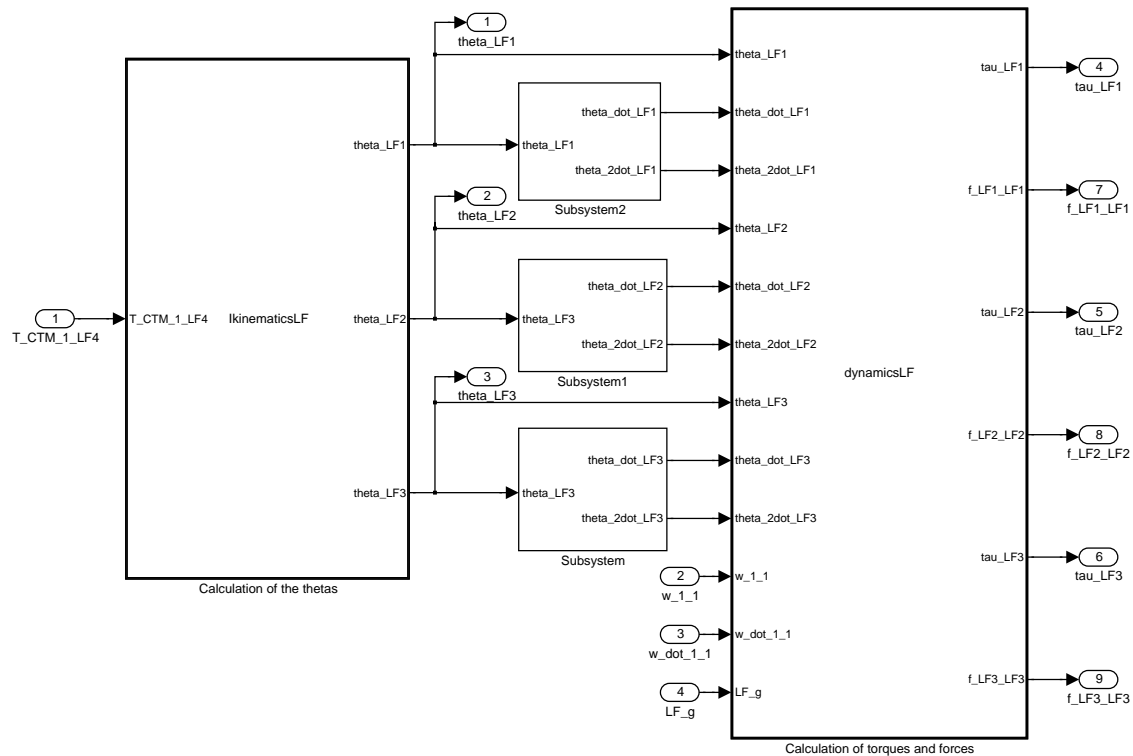


FIGURE 12.1: Subsystem for the LF leg.

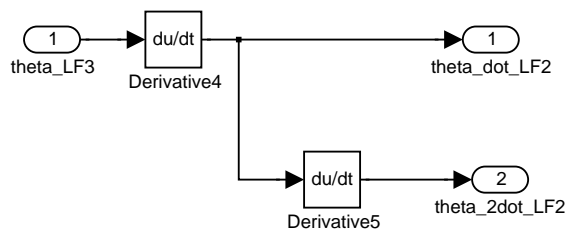


FIGURE 12.2: Contents of the subsystem Calculation of  $\theta_{\dot{LF}2}$  and  $\theta_{2\dot{LF}2}$  in Figure 12.1.

The two *Derivative* blocks are differentiating the input with respect to time, this gives the velocity  $\theta_{dot\_LF2}$  and acceleration  $\theta_{2dot\_LF2}$ .

The tests of the model of the LF leg were conducted as four different test scenarios; one with different constant inputs, and three with a pulse train on one input and constants on the others. The pulse train used as input is shown in Figure 12.3.

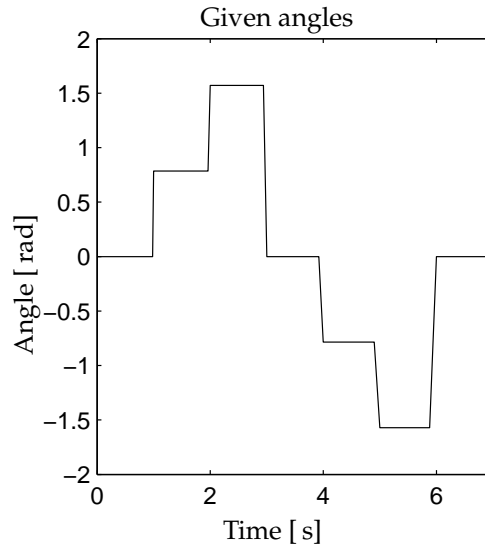


FIGURE 12.3: An illustration of the pulse train used as variable input.

The tests performed in this section are made in Simulink<sup>®</sup> with a *fixed-step size* of 0.1 [s] which give a samplings frequency of 10 [Hz] and this solver *ode4* (Runge-Kutta).

The test procedures and the results are described in appendix E on page 153 and only two will be described here. These two are the test with constant inputs and the test with the pulse train as input on  $\theta_{LF1}$ .

In the tests with constant inputs the angles were expected to be calculated with a precision of at least 0.01 [rad]. The input changes were also expected to influence the required torque for all the joints, but there were not expected any accelerations because the values were taken after the system had settled and were in steady state. Table 12.1 contains the results of the tests. Angles are donated  $\theta$  and torque are donated  $\tau$ . Pair is the LP supporting the weight of the body, and Leg Pair 3 (LP3) is for when the weight of the NBT is supported by all four legs.

The given angles for the inputs  $\pm\pi/2$  and  $\pm\pi/4$  are calculated to  $\pm 1.571$  and  $\pm 0.7854$ , respectively, which are correct. Therefore the precision for the calculation of the angles is at least 0.001 which fulfill the expected precision. The required torque are changing when the angles are changing, and the changes in the angle are also influencing the required

---

Input				Output					
$\theta_{LF1}$	$\theta_{LF2}$	$\theta_{LF3}$	LP	$\theta_{LF1}$	$\theta_{LF2}$	$\theta_{LF3}$	$\tau_{LF1}$	$\tau_{LF2}$	$\tau_{LF3}$
0	0	0	LP1*	0	0	0	-0.0104	-0.0107	0.0009
0	0	0	LP2	0	0	0	-0.1245	-0.0107	-0.1436
0	0	0	LP3	0	0	0	-0.1305	-0.0107	-0.1436
$\pi/4$	0	0	LP1	0.7854	0	0	0.1560	-0.0107	0.0267
$\pi/2$	0	0	LP1	1.571	0	0	0.2103	-0.0107	0.0369
$-\pi/4$	0	0	LP1	-0.7854	0	0	-0.1414	-0.0107	-0.0254
$-\pi/2$	0	0	LP1	-1.571	0	0	-0.2103	-0.0107	-0.0369
0	$\pi/4$	0	LP1	0	0.7854	0	0.0083	0.0818	0.0012
0	$-\pi/4$	0	LP1	0	-0.7854	0	0.0083	-0.0970	0.0013
0	$-\pi/2$	0	LP1	0	-1.571	0	0.0035	-0.1264	0.0021
0	0	$\pi/4$	LP1	0	0	0.7854	0.0362	-0.0099	0.0267
0	0	$\pi/2$	LP1	0	0	1.571	0.0463	-0.0077	0.0369

**TABLE 12.1:** Results from applying constant values as joint angles for the LF leg. \* When the angles for the leg joints all are zero is the leg in a position where it is supporting the weight of the body, but to test the difference in torque when supporting the weight and not supporting the weight is this test included.



torque for the other joints as expected. The required torques are as expected larger when the leg is supporting the weight of the body, which is for LP2 and LP3.

In the tests with a pulse train as input to  $\theta_{LF1}$ , were the calculated angles expected to follow their input. Changes in  $\theta_{LF1}$  is expected to give an increase in the required torque, which will accelerate the leg, and make it start moving. This will be followed by a decrease in the required torque, which will decelerate the leg and stop it at a certain position. The torque at this level is the torque required to hold the angle  $\theta_{LF1}$ . The direction  $\theta_{LF1}$  changes is in the direction of the acceleration and deceleration. The required torque is expected to be largest for  $\{LF1\}$  because this is the first joint. The torque for  $\tau_{LF2}$  is only expected to be influenced slightly when  $\theta_{LF1}$  is changed because it is perpendicular to the rotation direction.  $\{LF3\}$  is expected to have similar changes as  $\{LF1\}$ , just smaller. The results of applying the pulse train as input of  $\theta_{LF1}$  is illustrated on Figure 12.4. The blue line is for  $\{LF1\}$ , the magenta line is for  $\{LF2\}$  and the black line is for  $\{LF3\}$ .

The calculated angles for  $\theta_{LF1}$  follows the given input and the calculated angles for the other two inputs is a constant zero as expected. When  $\theta_{LF1}$  is changing the torque for  $\tau_{LF1}$  increases then decreases and finally returns to a level where it is stable. This level depends on the given angle, as it should. Changes in  $\tau_{LF2}$  are only minor and  $\tau_{LF3}$  follows  $\tau_{LF1}$  as expected.

The results of applying the pulse train as input to  $\theta_{LF2}$  and  $\theta_{LF3}$  were akin to the results for  $\theta_{LF1}$ . These results can as earlier mentioned be found in appendix E on page 153.

The tests gave the expected results, therefore is the tendencies of the model accepted. The model can be fully validated by comparing the model with the physical NBT, this is not at disposal but can be an advantage to do in future work.

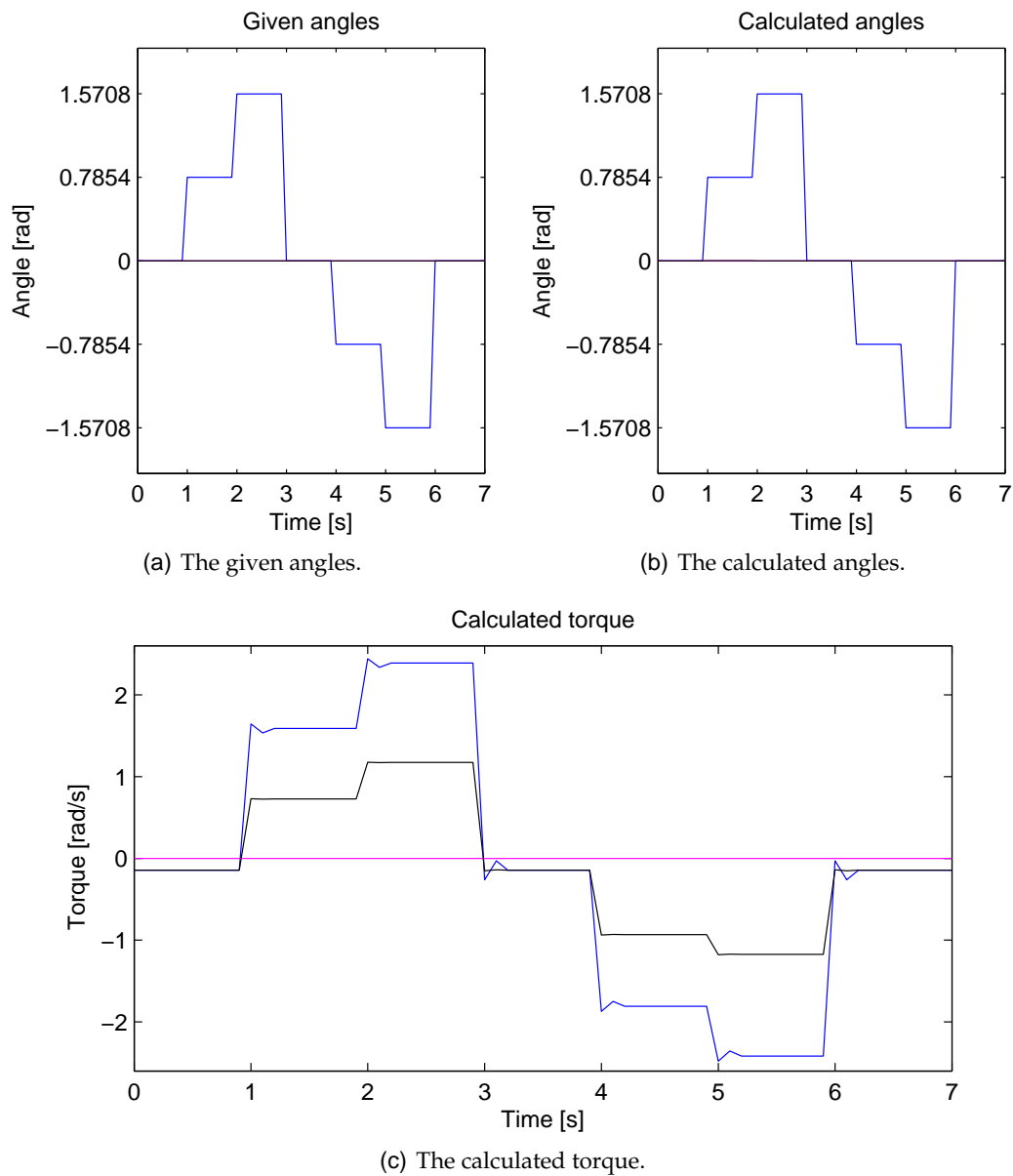
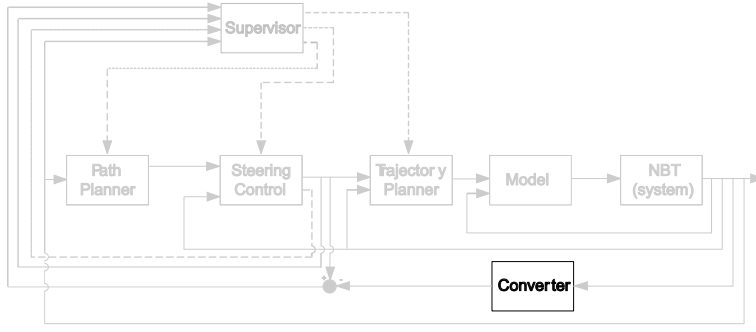


FIGURE 12.4: Test results for applying the pulse train as input to  $\theta_{LF1}$ . The blue line is for  $\{LF1\}$ , the magenta line is for  $\{LF2\}$  and the black line is for  $\{LF3\}$ .



## SENSOR MODELING

*This chapter describes how the feed back signals will be calculated and how these calculations are implemented in MATLAB<sup>®</sup> and Simulink<sup>®</sup>.*

The system combining described in Chapter 7 uses the following feedback signals;  $\bar{\theta}_{N,m}$ ,  $\bar{\dot{\theta}}_{N,m}$  and  $\bar{\ddot{\theta}}_{N,m}$ , for the *Steering control*, *Trajectory planner* and *Manipulator dynamics model*, these will be modeled in section 13.1. The converter shall calculate the position vectors between the shoulders and feet, from the obtained joint angles, these are used by the *Path planner* and *Supervisor*. In the verification of the inverse manipulator kinematics model in Chapter 10 is a function implemented to do these calculations, therefore this function will be used as the converter.

### 13.1 ROTATIONAL DYNAMICS

The rotational dynamics build on the analysis made in section 4.1. There will be limited from the forces generated by collision and the sensors will be kept as simple as possibly, as described in Chapter 8.

The NBT can be seen as an inverted pendulum, that only is at rest when the CM is straight over the line of support, else is the NBT falling, as illustrated on Figure 13.1.

$\vartheta_{CM}$  is the angle between vertical and the line between the supporting feet and CM.  $F_{g1}$  is the gravity force acting on the NBT, this is calculated in Equation (4.5a). The gravity can be converted into an angular acceleration  $\alpha$  by converting  $F_{g1}$  into an angular force and applying the formulas in Equation (13.1), this gives Equation (13.2). The friction between the feet and floor are assumed to be so large that the feet not slip when falling, therefore frictions will be disregarded in the model of the inverted pendulum. [SB00, chapter 13.4 p. 402-406]

$$F = m \cdot a \quad (13.1a)$$

$$a = r_{CM} \cdot \alpha \quad (13.1b)$$

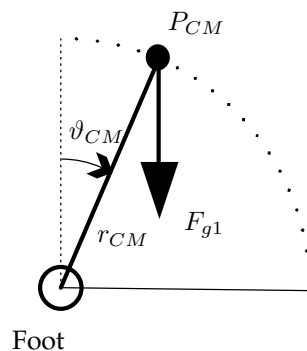


FIGURE 13.1: An illustration of how the gravity force is increasing the angular accelerating.

$$\begin{aligned}
 \alpha &= \ddot{\vartheta}_{CM} \\
 &= \frac{\frac{m_N \cdot g \cdot \sin(\vartheta_{CM})}{m_N}}{r_{CM}} \\
 &= \frac{g \cdot \sin(\vartheta_{CM})}{r_{CM}}
 \end{aligned} \tag{13.2}$$

$r_{CM}$  is the distance from the line between supporting feet to the CM, as illustrated on Figure 13.1.  $\vartheta_{CM}$  is the angle from vertical to  $r_{CM}$ , this is also illustrated the mentioned figure. The displacement of the CM from the line between the supporting feet will be used to find  $\vartheta_{CM}$ , this length will be denoted  $L_{CM}$ , an example of  $L_{CM}$  is illustrated on Figure 13.2.

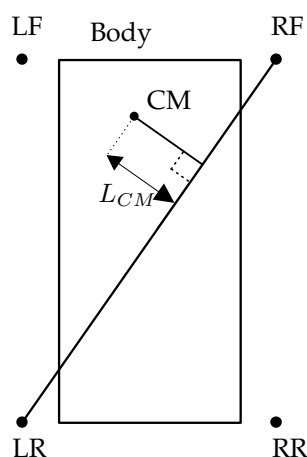


FIGURE 13.2: An illustration of how the distance between the CM and the line between the supporting feet is measured.

$L_{CM}$  will be found by making Right Angle Triangle (RAT) where the angles can be

found as illustrated on Figure 13.3(a) and use these angles to find  $L_{CM}$  in another RAT as illustrated on Figure 13.3(b).

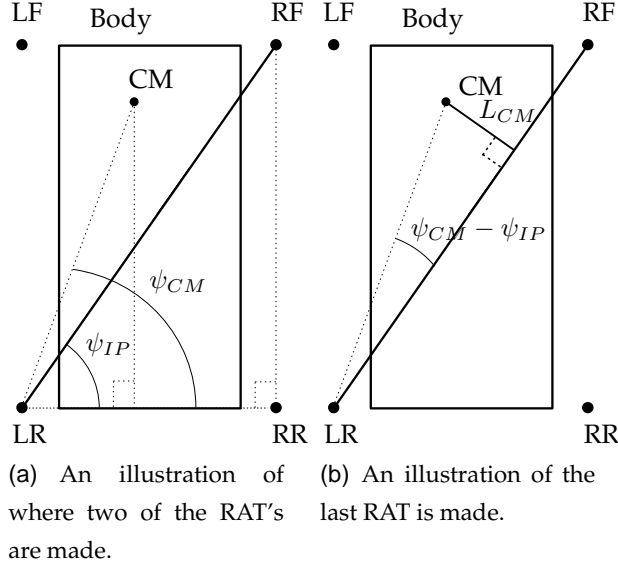


FIGURE 13.3: Illustrations of how the RAT is made to calculate  $L_{CM}$ .

$L_{CM}$  is calculated in Equation (13.3) with the supporting legs as above, the opposite solution is similar just with the opposite LP as supporting. The position vectors are between the shoulders and feet, they have to be changed to have the same origin when calculating  $L_{CM}$ . This can be done by use of the transformation matrices derived in Chapter 9, or adding with the distances in the current directions. The second solution is chosen because it will use less computations in the final implementation. To make the documentation of the equations more understandable will they be given in relation to coordinate system  $\{1\}$ , and the lengths will be used in the implementation. The placement of the coordinate system is illustrated on Figure 13.4.

$$\begin{aligned}\psi_{IP} &= \arctan \left( \frac{|P_{RF,x}| + |P_{LR,x}|}{|P_{RF,y}| + |P_{LR,y}|} \right) \\ \psi_{CM} &= \arctan \left( \frac{P_{CM,x} + |P_{LR,x}|}{-P_{CM,y} + |P_{LR,y}|} \right) \\ L_{CM} &= \sin(\psi_{CM} - \psi_{IP}) \sqrt{((P_{CM,x} + |P_{LR,x}|)^2 + (P_{CM,y} + |P_{LR,y}|)^2)} \quad (13.3)\end{aligned}$$

$H_{CM}$  is the height from the supporting feet to the CM along the z axis fo coordinate system  $\{1\}$ . There are used the magnitude of the position vectors components because the length from coordinate system  $\{1\}$  to the foot always is positive. The sign of  $P_{CM}$  tels where the CM is located in relation to the rear supporting leg. In the calculation with

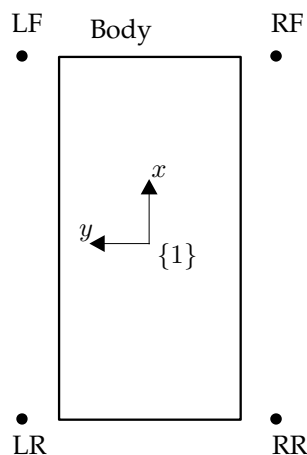


FIGURE 13.4: An illustration of the placement of coordinate system  $\{1\}$ .

the opposite LP as supporting changes the sign of  $P_{CM,y'}$  because the calculation is from the RR leg instead and the length therefore will be measured in the opposite direction. If the shoulders have different height from the floor, shall the height be corrected so it is correct for the CM, this is calculated in Equation (13.4).

$$H_{CM} = P_{LR,z} + (P_{LR,z} - P_{RF,z}) \cdot \frac{\tan(\psi_{CM} - \psi_{IP}) \cdot L_{CM}}{\sqrt{(|P_{RF,y}| + |P_{LR,y}|)^2 + (|P_{RF,x}| + |P_{LR,x}|)^2}} \quad (13.4)$$

With  $H_{CM}$  and  $L_{CM}$  known can  $r_{CM}$  be calculated by making a RAT as illustrated on Figure 13.5.

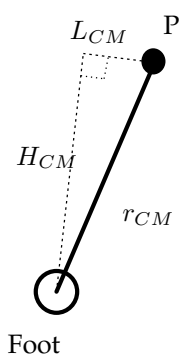


FIGURE 13.5: An illustration of how  $H_{CM}$  and  $L_{CM}$  is used to make a RAT to calculate  $r_{CM}$ .

$r_{CM}$  is calculated in Equation (13.5).

$$r_{CM} = \sqrt{L_{CM}^2 + H_{CM}^2} \quad (13.5)$$

$\vartheta_{CM}$  is calculate in Equation (13.6).

$$\vartheta_{CM}(i+1) = \arctan\left(\frac{L_{CM}}{H_{CM}}\right) + \theta_B(i) \quad (13.6)$$

$\theta_B(i)$  is the body angle at the former time step, this is calculated from the rotations around the coordinate systems axis by Equation (13.7).  $\theta_{B,x}(i)$  and  $\theta_{B,y}(i)$  is rotations around the x and y axis, respectively.

$$\theta_B(i) = \frac{\theta_{B,y}(i)}{\cos(\psi_{IP})} + \frac{\theta_{B,x}(i)}{\sin(\psi_{IP})} \quad (13.7)$$

Now all the parameters in Equation (13.2) are found and the angular acceleration around the supporting feet can be calculated by Equation (13.8). The derived formulas for  $H_{CM}$ ,  $L_{CM}$ ,  $\psi_{IP}$  and  $\psi_{CM}$  will not be inserted into the formula to keep the clarity in the Equation.

$$\begin{aligned} \ddot{\vartheta}_{CM}(i+1) &= \frac{g \cdot \sin(\vartheta_{CM}(i+1))}{r_{CM}} \\ &= \frac{g \cdot \sin(\arctan(\frac{L_{CM}}{H_{CM}}) + \theta_B(i))}{r_{CM}} \\ &= \frac{g \cdot \sin(\arctan(\frac{L_{CM}}{H_{CM}}) + \theta_B(i))}{\sqrt{(L_{CM}^2 + H_{CM}^2)}} \\ &= \frac{g \cdot \sin(\arctan(\frac{L_{CM}}{H_{CM}}) + \frac{\theta_{B,y}(i)}{\cos(\psi_{IP})} + \frac{\theta_{B,x}(i)}{\sin(\psi_{IP})})}{\sqrt{(L_{CM}^2 + H_{CM}^2)}} \end{aligned} \quad (13.8)$$

The acceleration is assumed to be constant between the samples, therefore can the angular velocity  $\dot{\vartheta}_{CM,f}$  and angle  $\vartheta_{CM,f}$  be calculated by Equation (13.9) and (13.10). [SB00, chapter 10 p. 296-298]

$$\dot{\vartheta}_{CM}(i+1) = \dot{\vartheta}_{CM}(i) + \ddot{\vartheta}_{CM}(i+1) \cdot t_s \quad (13.9)$$

$$\vartheta_{CM}(i+1) = \vartheta_{CM}(i) + \dot{\vartheta}_{CM}(i) \cdot t_s + \frac{1}{2} \cdot \ddot{\vartheta}_{CM}(i+1) \cdot t_s^2 \quad (13.10)$$

$\vartheta_{CM}$ ,  $\dot{\vartheta}_{CM}$  and  $\ddot{\vartheta}_{CM}$  are rotated around the line between the supporting feet, but the sensor outputs for  $\bar{\theta}_{N,m}$ ,  $\dot{\bar{\theta}}_{N,m}$  and  $\ddot{\bar{\theta}}_{N,m}$  shall be around the axis of coordinate system  $\{1\}$ , therefore have the results to be transformed. The falling will not influence the rotation around the z axis, this will turning on the other hand but since the NBT shall walk along a line as defined in Chapter 2, the rotations around the z axis not will be modeled. The

rotations are converted to coordinate system  $\{1\}$  in Equation (13.11).

$$\ddot{\theta}_{N,m,x} = \ddot{\vartheta}_{CM}(i+1) \cdot \sin(\psi_{IP}) \quad (13.11a)$$

$$\ddot{\theta}_{N,m,y} = \ddot{\vartheta}_{CM}(i+1) \cdot \cos(\psi_{IP}) \quad (13.11b)$$

$$\dot{\theta}_{N,m,x} = \dot{\vartheta}_{CM}(i+1) \cdot \sin(\psi_{IP}) \quad (13.11c)$$

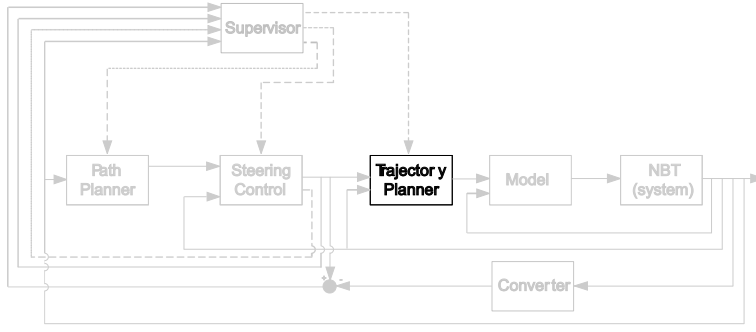
$$\dot{\theta}_{N,m,y} = \dot{\vartheta}_{CM}(i+1) \cdot \cos(\psi_{IP}) \quad (13.11d)$$

$$\bar{\theta}_{N,m,x} = \vartheta_{CM}(i+1) \cdot \sin(\psi_{IP}) \quad (13.11e)$$

$$\bar{\theta}_{N,m,y} = \vartheta_{CM}(i+1) \cdot \cos(\psi_{IP}) \quad (13.11f)$$

(Note: The sensor is implemented as a function, this can be found  
in /simulations/control/sensor\_angle.m)





## TRAJECTORY PLANNER

*This chapter contains the design of the trajectory planner. The trajectory planner is calculating trajectories for the feet in the swing-, stand- and start walk phase.*

The construction of the trajectory planner will be based on the analysis of the CPG made in Chapter 5 and the analysis of quadruped walk made in Chapter 4. The inputs and outputs to the trajectory planner are described in the system combining in Chapter 7, the trajectory planning part is illustrated on Figure 14.1.

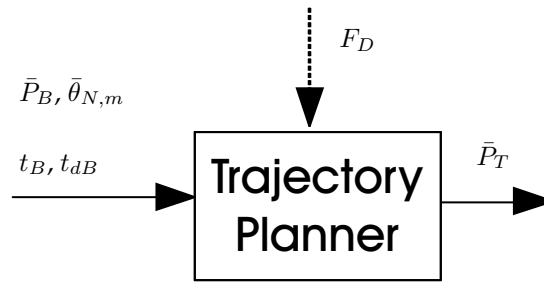


FIGURE 14.1: An illustration of the trajectory planner with inputs and outputs. The illustration is a part of the system diagram illustrated on Figure 7.1.

The variables on the figure have all been introduced earlier, but the meaning of them briefly will be introduced again.  $\bar{P}_B$  are the position vectors from the steering control for where the feet shall be placed on the floor, this position can be changed on the fly by the steering control, therefore shall the trajectory planner correct the trajectories for the feet, this will be discussed doing the design.  $F_D$  is a time discrete event triggering the phase of the walk, these phases are; start walk or which leg there are in the swing phase.  $t_{dB}$  is the period time and  $t_B$  is the time used.  $\bar{\theta}_{N,m}$  is the measured rotation of the body around the longitudinal, latitudinal and vertical axis, these are illustrated on Figure 3.8, these rotations will be used to calculate the new positions of the feet. These positions are the output from the trajectory planner, they are donated  $\bar{P}_T$ .

The documentation of the trajectory planner will be divided into two sections, one for

the walking and one for the start walk, the walking will be in section 14.1 and the start walking will be in section 14.2. After the design of the trajectory planner is documented will it be structural tested in section 14.3.

## 14.1 WALKING

The walk in trot is analysed in Chapter 4, here is the curve for the swing and stand phase discussed. The swing phase shall follow the parabolic curve in Equation (4.1), which is defined in a plan perpendicular to the floor, where  $y$  is the height that the foot is lifted from the floor.

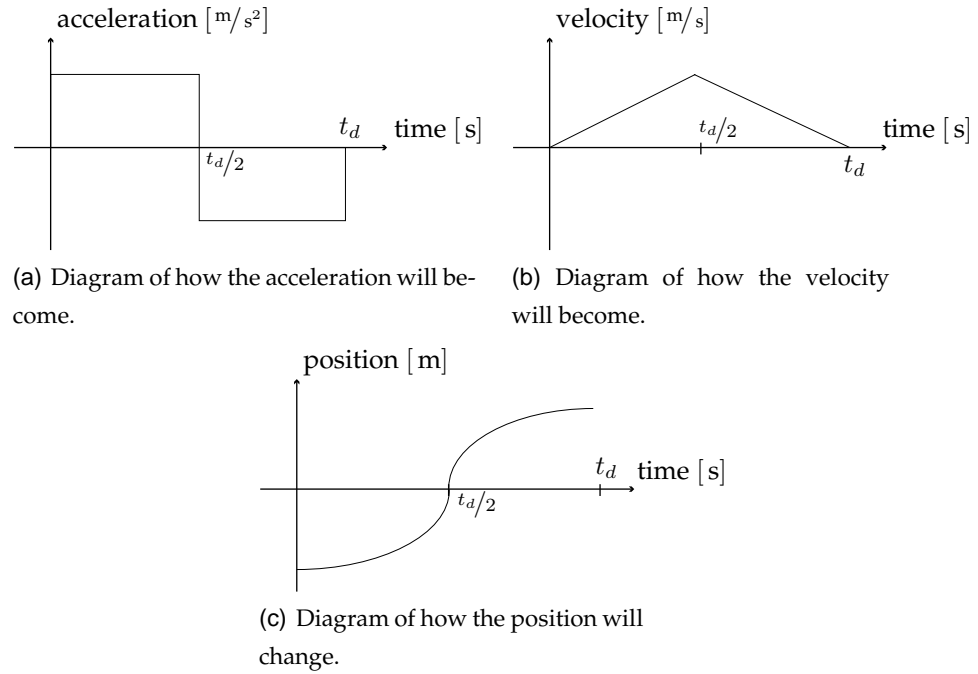
The stand phase shall always have the foot on the floor. The movement of the feet are started in the beginning and stopped in the end of the trajectories for the feet, as the states  $A$ ,  $B$ ,  $C$  and  $E$  for the state machine in Chapter 5, describes. Therefore are the velocity zero in the start and end of the sequences, the feet have to be accelerated and decelerated to start and stop the movement of them. The acceleration and deceleration are desirable to keep as small as possibly, because the influence on the rotational velocity of the body, then will be minimized. Therefore are the acceleration and deceleration time each chosen to half of the period time, with a constant acceleration, this will make step as Figure 14.2(a) illustrates. The stepping can be avoided by ramping the acceleration from zero to the desired, this will make the acceleration of the foot more smooth, but it is the joint accelerations there are interesting to make smooth and they are not linear dependent on the foot movement. Therefore the method with constant accelerations will be implemented, it can be changed if the joint accelerations are unacceptable. The velocity with this method will be as Figure 14.2(b) illustrates and the position as Figure 14.2(c).

The period time  $t_{dB}$  is given by the *steering control*, and can be changed on the fly, if the velocity or direction of the movement is changed. Therefore has the needed acceleration to be calculated at every step, the acceleration will be in the direction of  $L_{SW}$  for both the swing and stand phase,  $L_{SW}$  is illustrated on Figure 14.3 for the swing phase.

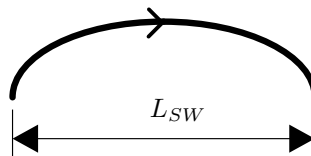
If the joint accelerations in the swing phase is unacceptable can the acceleration for this phase be changed so it is along the curve. The acceleration is calculated in Equation (14.1).

$$a_i = 2 \cdot \frac{(L_{SW}/2)}{(t_{dB}/2)^2} \quad (14.1a)$$

$$a_f = -a_i \quad (14.1b)$$



**FIGURE 14.2:** *Schematics of the acceleration, velocity and position change with constant acceleration and deceleration in the first and second half of the period time, respectively.*



**FIGURE 14.3:** *An illustration of where  $L_{SW}$  is in the swing phase.*

$a_i$  is the acceleration in the start of the period and  $a_f$  is the deceleration in the end of the period.

When the steering controller is changing the position vector to avoid falling can  $L_{SW}$  be changed, therefore has it also to be calculated for every step. The direction of  $L_{SW}$  on the floor can also be changed, this makes the new  $L_{SW}$  go in another direction then the old  $L_{SW}$  did. An example of this is illustrated on Figure 14.4, where  $\bar{P}_S$  is the position vector at the start of the period,  $\bar{P}_P$  and  $\bar{P}_B$  are the position vectors given by the path planner and the corrected one from the steering control, respectively.  $\bar{P}_T(i-1)$  is the position vector calculated at the former trajectory point. The old  $L_{SW}$  is from  $\bar{P}_S$  to  $\bar{P}_P$  and the new  $L_{SW}$  is through  $\bar{P}_T(i-1)$  to  $\bar{P}_B$ .

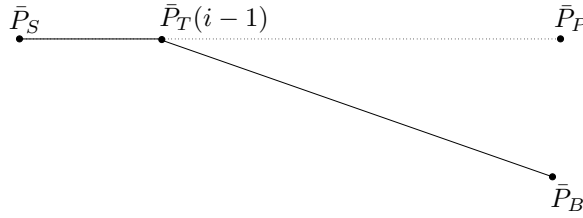


FIGURE 14.4: An example of how the direction of  $L_{SW}$  can be changed when the steering control is correcting to avoid falling.

The example demonstrates that the new  $L_{SW}$  depends on  $\bar{P}_T(i-1)$  and  $\bar{P}_B$ . In the swing phase can  $L_{SW}$  be derived from the Equation (4.1) because it is the only unknown parameter, but in the stand phase can this method not be used, therefore will  $L_{SW}$  not be calculated with this method. Instead  $L_{SW}$  will be calculated with use of the length from  $\bar{P}_T(i-1)$  to  $\bar{P}_B$  and the acceleration in  $\bar{P}_T(i-1)$ , this is calculated in Equation (14.2). The equation is splitted in two because the acceleration can be zero in the start but not in the end, therefore has this form been chosen. The length from  $\bar{P}_T(i-1)$  to  $\bar{P}_B$  is donated  $l_f$  and is calculated by use of Pythagoras in Equation (14.3).

$$L_{SW} = \begin{cases} l_f + t^2 \cdot a_i \cdot 1/2 & \text{for } t < t_{dB}/2 \\ 2 \cdot \left(\frac{t_{dB}}{2}\right)^2 \cdot a_f \cdot 1/2 & \text{else} \end{cases} \quad (14.2)$$

$$l_f = \sqrt{((P_{B,z} \cdot \cos(\theta_{Nm,y}) - P_{T,z}(i-1) \cdot \cos(\theta_{Nm,y}))^2 + (P_{B,y} \cdot \cos(\theta_{Nm,z}) - P_{T,y}(i-1) \cdot \cos(\theta_{Nm,z}))^2)} \quad (14.3)$$

$P_{B,y}$ ,  $P_{B,z}$ ,  $P_{T,y}(i-1)$  and  $P_{T,z}(i-1)$  are the  $y$  and  $z$  component of the position vector for  $P_B$  and  $P_T$ , respectively.  $\theta_{Nm,y}$  and  $\theta_{Nm,z}$  are the angles of the body relatively to the floor for the  $y$  and  $z$  axis of  $\{1\}$ , respectively. The angles are used to project the components of the position vectors into a plan parallel with the floor.

The position of the foot in the direction of the trajectory for the foot is donated  $x$  and is calculated in Equation (14.4).

$$x = \begin{cases} t^2 \cdot a_i \cdot 1/2 & \text{for } t < t_{dB}/2 \\ L_{SW} + (t_{dB} - t)^2 \cdot a_f \cdot 1/2 & \text{else} \end{cases} \quad (14.4)$$

The height the feet shall be lifted in the swing phase can new be calculated by Equation (4.1). The height the feet are lifted at the middle of the swing phase will be found in Chapter 15, together with the other limits for the walk. The  $x$  and  $y$  axis are used to calculate the components of the position vector that is the output from this part, this is done in Equation (14.5).

$$\bar{P}_T = [P_{T,x} \ P_{T,y} \ P_{T,z}]^T \quad (14.5a)$$

$$P_{T,x} = \begin{cases} P_{B,x} - y \cdot \cos(\omega_y) \cdot \cos(\omega_z) - (L_{SW} - x) \cdot \sin(\psi_T) & \text{for swing} \\ P_{B,x} - (L_{SW} - x) \cdot \sin(\psi_T) & \text{for stand} \end{cases} \quad (14.5b)$$

$$P_{T,y} = P_{B,y} + (L_{SW} - x) \cdot \cos(\rho_T) \cdot \cos(\omega_z) \quad (14.5c)$$

$$P_{T,z} = P_{B,z} + (L_{SW} - x) \cdot \sin(\rho_T) \cdot \cos(\omega_y) \quad (14.5d)$$

If the position vectors  $P_B$  and  $P_O$  have different  $x$  components is the height of the foot chancing, this difference is gives a angle  $\psi_T$ , an example for the RF foot is illustrated on Figure 14.5. The figure is for the swing phase, but is similar for the stand phase.

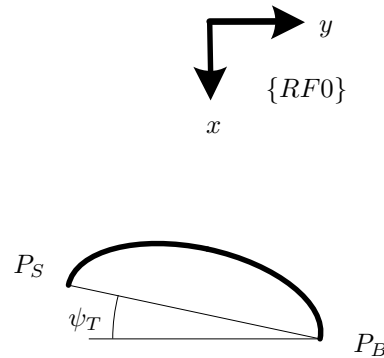


FIGURE 14.5: An illustration of where the angle  $\psi_T$  is. The curve between  $P_S$  and  $P_B$  is the ballistic curve for the foot.

$\rho_T$  is the angle between the body of the NBT and direction the foot is moved, an example for the RF leg is illustrated on Figure 14.6.

$\bar{P}_T(i - 1)$  is not an input to the trajectory planner in the control system architecture, therefore has the architecture to be modified. When the architecture is changed it will be

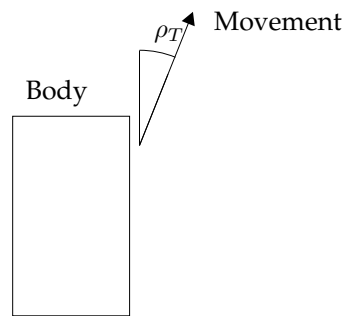


FIGURE 14.6: An example of how the angle  $\rho_T$  is calculated.

advantageous to use  $\bar{P}_m$  as  $\bar{P}_T(i - 1)$  because the trajectory planner then get a feedback of the obtained position instead of the position there were desired to have obtained. This make the trajectory planner correct the output error of the system and make it converge against zero, thereby is the trajectory planner working as an feedback controller with varying parameters, dependent on the time and phase of the walk. The closer  $t_B$  is to  $t_{dB}$  the faster is the converging. In the stand phase is it converging straight for  $\bar{P}_B$ , but in the swing phase the foot will be lifted, thereby giving a bending on the converging.

The velocity and acceleration in the joints are already calculated in the implementation of the model used in the structural model test in Chapter 12. The velocities and accelerations are calculated by differentiating the angles over the time. This could also have been done by use of *Jacobian* which in the field of robots generally are used to relate Cartesian velocities to joint velocities [Cra05, section 5.7 p. 149-151]. The method to calculate the velocities and accelerations will be kept as it is, because the implementation already is tested. A new method to calculate the joint accelerations and velocities will require a new test of the model.

The trajectory planner is implemented as a MATLAB<sup>®</sup> function calculating the position vector for all the legs for the trajectory point. (Note: The function can be found in `/simulations/control/trajectory_calc.m`.)

## 14.2 START WALK

The start walking is the state *A. Start walk* in the state machine in Chapter 5, where the feet in LP2 are lifted from the floor as Figure 5.4 illustrates and the feet of LP1 are kept at the same position as they were at the start. The objectives state that the NBT shall be able to start walking from standing, but the design and implementation shall be kept simple.

The foot will be lifted from start position on the floor to the position demanded by

the path planner, this position will normally be at the middle of the curve for the swing phase. Acceleration and deceleration will be disregarded to simplify the design and because the method to implement it already is described in section 14.1, instead the movement will be made with a even distribution of the length between the steps there shall be moved to the time  $t_{dB}$ . The output position vector from this part, is calculated in Equation (14.6).

$$\bar{P}_T = [P_{T,x} \ P_{T,y} \ P_{T,z}]^T \quad (14.6a)$$

$$P_{T,x} = \frac{(P_{B,x} - P_{m,x})}{(t_{dB} - (t_B - t_s))} \cdot t_s + P_{O,x} \quad (14.6b)$$

$$P_{T,y} = \frac{(P_{B,y} - P_{m,y})}{(t_{dB} - (t_B - t_s))} \cdot t_s + P_{m,y} \quad (14.6c)$$

$$P_{T,z} = \frac{(P_{B,z} - P_{m,z})}{(t_{dB} - (t_B - t_s))} \cdot t_s + P_{m,z} \quad (14.6d)$$

$t_s$  is the step time, which earlier is chosen to 0.1 [s]. The start walking sequence will be implemented together with the walking sequence in the function.

## 14.3 TEST

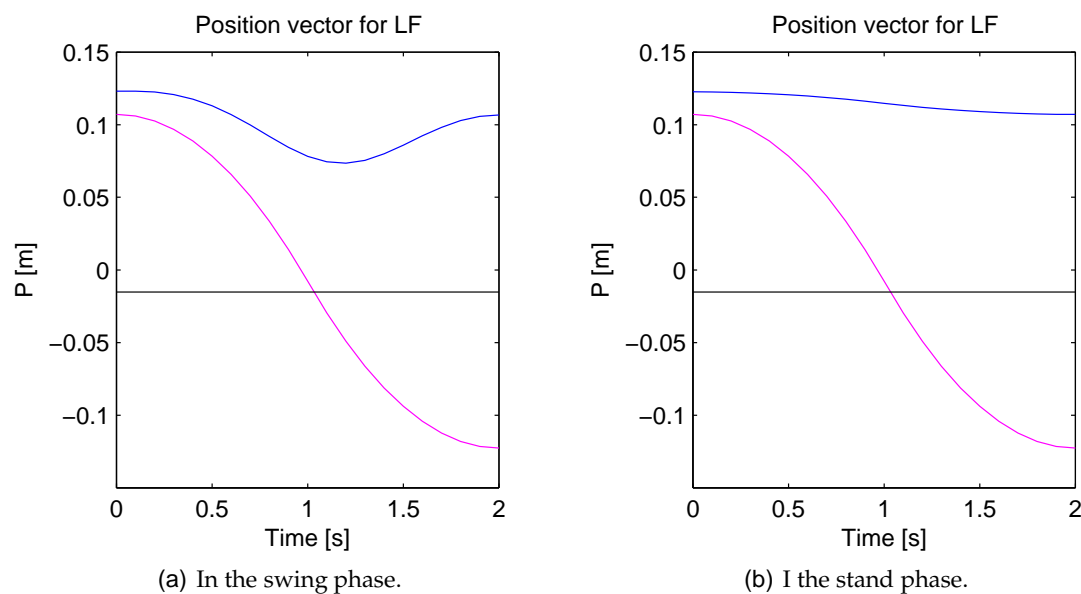
The implemented function to plan the trajectory is structurally tested to find out if the calculated trajectory is correct. The test is performed for all the legs, but only the results for the LF foot will be documented, the other results can be generated with this MATLAB<sup>®</sup> script `/simulation/control/trajectoryTEST.m`.

The test is performed by letting the function calculate a trajectory between  $P_S$  and  $P_B$ , which has the values in Equation (14.7). These are calculated by the converter with the angles  $-0.1$  [rad] and  $0$  [rad] for  $\theta_{LF2}$  and  $\theta_{LF3}$ , respectively.  $\theta_{LF1}$  has the value  $\pi/4$  [rad] and  $-\pi/4$  [rad] for  $P_S$  and  $P_B$ , respectively. The difference in the first two components of the  $P_S$  and  $P_B$  is due to the displacement in the knee.

$$P_S = [0.1226 \ 0.1071 \ -0.0152]^T \quad (14.7a)$$

$$P_B = [0.1071 \ -0.1226 \ -0.0152]^T \quad (14.7b)$$

The tests are performed over 2[s] with  $t_s$  as 0.1[s]. The calculated trajectory for the swing phase and stand phase is illustrated in Figure 14.7(a) and Figure 14.7(b), respectively. The blue line is for  $P_{T,x}$ , the magenta for  $P_{T,y}$  and the black line is for  $P_{T,z}$ .



**FIGURE 14.7:** The calculated trajectory for the LF leg between  $P_S$  and  $P_B$ .

The blue line is for  $P_{T,x}$ , the magenta line is for  $P_{T,y}$  and the black line is for  $P_{T,z}$ .

The curve for  $P_{T,x}$  decreases in the first half and increases in the last half of the swing phase, this is correct because the foot is lifted up from the floor in the first half and put down in the last half, the length from the shoulder to the foot decreasing when the foot is lifted. The  $P_{T,x}$  bends in the start and end of the stand phase, this is because the acceleration and deceleration are starting and stopping to a velocity of zero, therefore is the curve for  $P_{T,x}$  bending when compared to the time. Both tests starting in 0.1226 and ending in 0.1071 as the values of  $P_S$  and  $P_B$  defines.

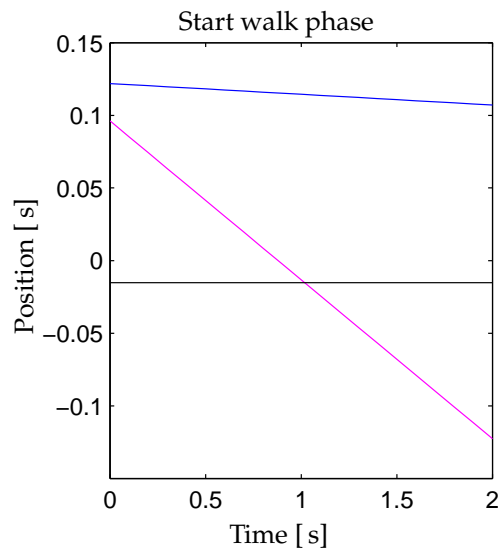
The curve for  $P_{T,y}$  is decreasing from  $P_{S,x}$  and  $P_{B,x}$  in both tests, with an accelerating in the first half and decelerating in the second half, as desired to minimize the influence of the impact with the floor. The curve for  $P_{T,z}$  is constant in both tests, this is correct because the position in this direction is constant in this test.

The trajectory calculations for the start walk phase is designed to calculate a trajectory between two points, with an even distributed step length and to  $t_{dB}$ , therefore it will be tested with the same inputs as the walk phase. The calculated trajectory is illustrated in Figure 14.8

The three curves are straight lines between  $P_S$  and  $P_B$ , this shows that the step length are even distributed and that the positions are calculated correct.

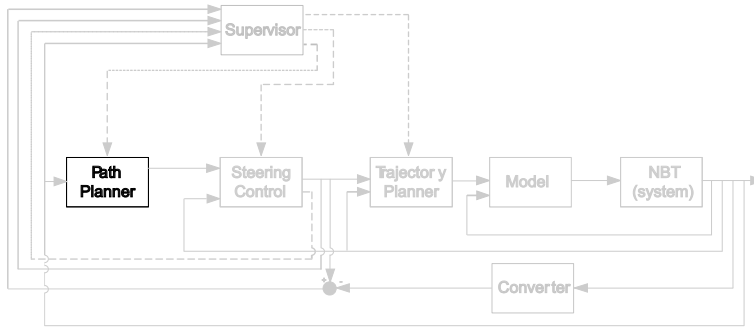
In future work could it be interesting to test the trajectories with a shift in in the  $\bar{P}_B$ , this is expected to give the same result as the tests above.





**FIGURE 14.8:** The calculated trajectory for the start walk phase for the LF leg, the trajectory is calculated between  $P_S$  and  $P_B$ . The blue line is for  $P_{T,x}$ , the magenta line is for  $P_{T,y}$  and the black line is for  $P_{T,z}$ .





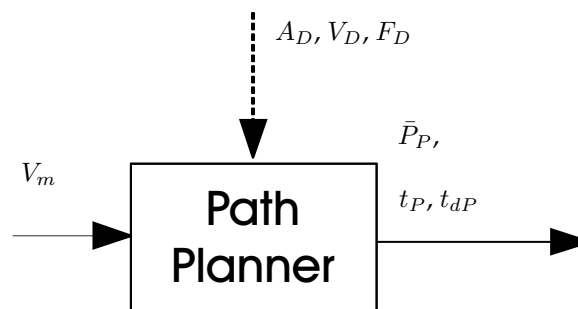
## PATH PLANNER

*This chapter contains the design of the Path planner. First will the claims to the part be introduced and afterwards will the area of movement doing walk be discussed, these areas will be used to chose some suited positions vectors for the end position in the phase of the walk. Finally is the period time derived.*

The construction of the path planner will as the trajectory planner be based on the analysis of the CPG made in Chapter 5 and the analysis of quadruped walk made in Chapter 4. The objectives state that the path planner shall be kept as simple as possibly because there already is made a path planner at the university of Newcastle, which is more sophisticated then the path planner in this project.

The path planner give the position vectors for where the feet shall be in the end of the swing, stand, and start walk phase, in relation to the coordinate systems locked to the shoulders, as defined in Chapter 10. It also control the times to get the feet to the desired position  $t_{dP}$  and the time used  $t_P$ .

The inputs and outputs to the path planner are described in the system combining in Chapter 7, the path planning part is illustrated in a rearranged form on Figure 15.1.



**FIGURE 15.1:** An illustration of the path planner with inputs and outputs.  
The illustration is a part of the system diagram illustrated on Figure 7.1.

The output from the path planner is a position vector for where each of the feet shall be when the current phase ends. These positions depends on the movement area for the legs,

---

this will be derived in the following. The angles of  $\theta_{LF2}$  and  $\theta_{LR2}$  is defined to 0.1 [rad] and  $-0.1$  [rad] for  $\theta_{RF2}$  and  $\theta_{RR2}$  in Chapter 4, these angles are chosen to increase the standing area when shifting legs in the swing and stand phase, thereby making the NBT more stable against disturbances as contact with other players etc.

The NBT will as in Chapter 13 be seen as an inverted pendulum when supported by two feet, the balance were found to be dynamics in Chapter 4. The acceleration is exponential increasing with  $\vartheta_{CM}$ , therefore the closer the center of pressure is to the line between the supporting feet, the smaller the acceleration of the body is, thereby the stabler it is. The stability of the body is a criteria for the trot locomotion, and give the possibility of faster motion because the body is more stable.

The point of pressure is a projection of the CM for the whole NBT onto the floor in the direction of the gravity. The movement of the legs in the swing and stand phase are changing the location of the CM doing the movement, but the CM for the body, hip, neck and head together are constant because the joint angles are held constant in trot, therefore will the movement area for the legs be placed in relation to this. The CM is calculated by use of the function for it made in the manipulator dynamics in Chapter 11, this place the CM at 0.0056 [m] on the x axis of coordinate system  $\{1\}$ . This is 0.0146 [m] from the middle between the front and rear shoulders in the direction of the front.

The unstableness of the inverted pendulum can be used to make the walk faster by getting the center of pressure to be in front of the support line, so the NBT would fall into the next step. The problem with this solution is that the NBT almost is two times longer then it is wide, therefore will the fall be more sideways then forward. This is a design decision for the steering controller. The angular acceleration of the fall is as earlier mentioned exponential increasing, therefore has the center of pressure to be placed close to the line of support to minimize the acceleration of the fall.

The position when the swing phase ends and the stand phase starts will be set to a rotation of  $-\pi/6$  [rad] for  $\theta_{LF1}$ ,  $\theta_{RF1}$ ,  $\theta_{LR1}$  and  $\theta_{RR1}$ , with the knees in the initial position as illustrated on Figure 15.2. The position vectors for the swing phase are shown in Equation (15.1).

$$\bar{P}_{P,LF} = [0.1462 \quad -0.0717 \quad 0.0152]^T \quad (15.1a)$$

$$\bar{P}_{P,RF} = [0.1462 \quad -0.0717 \quad -0.0152]^T \quad (15.1b)$$

$$\bar{P}_{P,LR} = [0.1462 \quad -0.0717 \quad 0.0152]^T \quad (15.1c)$$

$$\bar{P}_{P,RR} = [0.1462 \quad -0.0717 \quad -0.0152]^T \quad (15.1d)$$

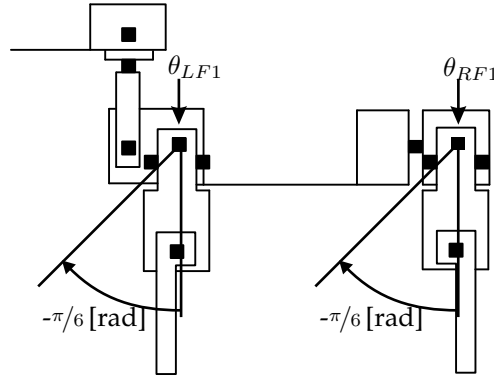


FIGURE 15.2: An illustration of the leg rotation when walking. The NBT is seen from its left side.

The position vectors for the stand phase is calculated so the middle of the trajectory for each of the feet is 0.0146 [m] in front of the shoulder, this give the position vectors in Equation (15.2) for the stand phases.

$$\bar{P}_{P,LF} = [0.1462 \ 0.0425 \ 0.0152]^T \quad (15.2a)$$

$$\bar{P}_{P,RF} = [0.1462 \ 0.0425 \ -0.0152]^T \quad (15.2b)$$

$$\bar{P}_{P,LR} = [0.1462 \ 0.0425 \ 0.0152]^T \quad (15.2c)$$

$$\bar{P}_{P,RR} = [0.1462 \ 0.0425 \ -0.0152]^T \quad (15.2d)$$

The center of pressure shall be in front of the line of support, but still close to it, this checked for the position vectors in Equation (15.1) and (15.2). This will be done at the phase shift because the position vectors for this position are known. The LF and RR leg will be in the swing phase and the RF and LR in the stand phase. The CM for the whole NBT is as earlier calculated by the function implemented in the manipulator dynamics in Chapter 11, this is using the joint angles as input, therefore is the position vectors converted to joint angles by the inverse manipulator kinematics derived in Chapter 10. An equation to calculate length between the line of support and center of pressure is derived in Equation (13.3). This is calculated in Equation (15.3). The equation is derived for the position vectors given in coordinate system  $\{1\}$ , but the position vectors derived in this chapter is given from the coordinate system locked to the shoulders, therefore are the lengths in the body added to the position vector components in the calculation. These are the lengths used in deriving of the kinematics model in Chapter 9, a table over the

---

distances can be found in Appendix A.

$$\begin{aligned}
\psi_{IP} &= \arctan \left( \frac{|\bar{P}_{RF,x}| + |\bar{P}_{LR,x}|}{|\bar{P}_{RF,y}| + |\bar{P}_{LR,y}|} \right) \\
&= \arctan \left( \frac{|\frac{0.1700}{2} - 0.0425| + |\frac{0.1700}{2} + \frac{0.0360}{2} + 0.0425|}{|\frac{0.1700}{2} + 0.0152| + |\frac{0.1700}{2} + 0.0152|} \right) \\
&= 0.9959 [\text{rad}] \\
\psi_{CM} &= \arctan \left( \frac{\bar{P}_{CM,x} + |\bar{P}_{LR,x}|}{\bar{P}_{CM,y} + |\bar{P}_{LR,y}|} \right) \\
&= \arctan \left( \frac{-0.0052 + |\frac{0.1700}{2} + \frac{0.0360}{2} + 0.0425|}{0 + |\frac{0.1700}{2} + 0.0152|} \right) \\
&= 0.8142 [\text{rad}] \\
L_{CM} &= \sin(\psi_{CM} - \psi_{IP}) \sqrt{(\bar{P}_{CM,x} + |\bar{P}_{LR,x}|)^2 + (\bar{P}_{CM,y} + |\bar{P}_{LR,y}|)^2} \\
&= \sin(0.8142 - 0.9959) \cdot \sqrt{(-0.0052 + |\frac{0.1700}{2} + \frac{0.0360}{2} + 0.0425|)^2} \\
&\quad + (0 + |\frac{0.1700}{2} + 0.0152|)^2} \\
&= -0.0137 [\text{m}]
\end{aligned} \tag{15.3}$$

$L_{CM}$  shall as defined above be positive and close to the line of support, this is not fulfilled with the chosen position vectors, therefore has the position vectors to be corrected.  $L_{CM}$  is the length to the CM and this can only be moved forward by moving the legs more forward, this will be done by decreasing the backwards movement of the legs in the stand phase defined in Equation (15.2). The movement will be changed from the 0.0425 [m] to 0.02 [m] in Equation (15.4).

$$\bar{P}_{P,LF} = [0.1462 \ 0.0200 \ 0.0152]^T \tag{15.4a}$$

$$\bar{P}_{P,RF} = [0.1462 \ 0.0200 \ -0.0152]^T \tag{15.4b}$$

$$\bar{P}_{P,LR} = [0.1462 \ 0.0200 \ 0.0152]^T \tag{15.4c}$$

$$\bar{P}_{P,RR} = [0.1462 \ 0.0200 \ -0.0152]^T \tag{15.4d}$$

$L_{CM}$  with the new position vector is calculated in Equation (15.5).

$$\begin{aligned}
 \psi_{IP} &= \arctan \left( \frac{|\bar{P}_{RF,x}| + |\bar{P}_{LR,x}|}{|\bar{P}_{RF,y}| + |\bar{P}_{LR,y}|} \right) \\
 &= \arctan \left( \frac{|\frac{0.1700}{2} - 0.0200| + |\frac{0.1700}{2} + \frac{0.0360}{2} + 0.0200|}{|\frac{0.1700}{2} + 0.0152| + |\frac{0.1700}{2} + 0.0152|} \right) \\
 &= 0.9959 [\text{rad}] \\
 \psi_{CM} &= \arctan \left( \frac{\bar{P}_{CM,x} + |\bar{P}_{LR,x}|}{\bar{P}_{CM,y} + |\bar{P}_{LR,y}|} \right) \\
 &= \arctan \left( \frac{-0.0052 + |\frac{0.1700}{2} + \frac{0.0360}{2} + 0.0200|}{0 + |\frac{0.1700}{2} + 0.0152|} \right) \\
 &= 1.0031 [\text{rad}] \\
 L_{CM} &= \sin(\psi_{CM} - \psi_{IP}) \sqrt{(\bar{P}_{CM,x} + |\bar{P}_{LR,x}|)^2 + (\bar{P}_{CM,y} + |\bar{P}_{LR,y}|)^2} \\
 &= \sin(1.0031 - 0.9959) \cdot \sqrt{((-0.0012 + |\frac{0.1700}{2} + \frac{0.0360}{2} + 0.0200|)^2 \\
 &\quad + (0 + |\frac{0.1700}{2} + 0.0152|)^2)} \\
 &= 0.0007 [\text{m}] \\
 &\approx 0 [\text{m}]
 \end{aligned} \tag{15.5}$$

The center of pressure is with these position vectors almost on the line of support, therefore  $\ddot{v}_{CM}$  will be minimized as desirable. Therefore will the position vectors in Equation (15.1) and (15.4) be used for the swing and stand phase, respectively.

In the swing phase shall the foot be lifted  $h$  from the floor as described in Chapter 4 and Chapter 14, this lift shall be large enough to secure that the foot is cleared from the floor so the NBT not stumbles and not to large so there are used unnecessary energy on the movement. The floor that the NBT will be walking on is plane, therefore will the foot only be lifted 0.01 [m] because this should be sufficient to clear the floor. The position vectors for the start walk then will be as in Equation (15.6).

$$\bar{P}_{P,LF} = [0.1362 \quad -0.0146 \quad 0.0152]^T \tag{15.6a}$$

$$\bar{P}_{P,RF} = [0.1462 \quad -0.0146 \quad -0.0152]^T \tag{15.6b}$$

$$\bar{P}_{P,LR} = [0.1462 \quad -0.0146 \quad 0.0152]^T \tag{15.6c}$$

$$\bar{P}_{P,RR} = [0.1362 \quad -0.0146 \quad -0.0152]^T \tag{15.6d}$$

The maximum period time the swing and stand phase may take to obtain the velocity

desired in Chapter 4 are calculated in Equation (15.7).

$$\begin{aligned}
 t_{dP} &= \frac{0.0717 + 0.0200 \text{ [m]}}{0.1467 \text{ [m/s]}} \\
 &= 0.6251 \text{ [s]} \\
 &\approx 0.6 \text{ [s]}
 \end{aligned} \tag{15.7}$$

The result is rounded toward zero because the movements thereby get a little faster then the minimum velocity desired. The phase between the legs in trot are constant, therefore are the time for a swing phase and stand phase the same. The initial value for the period time  $t_{dP}$  will be as in Equation (15.7), if the supervisor desires a higher velocity  $V_D$  of the NBT a new  $t_{dP}$  will be calculated with Equation (15.8).

$$t_{dP} = \frac{d}{V_D} \tag{15.8}$$

$d$  is the step length there shall be moved in the period. The time used in the period  $t_P$  is reset when the phase is changed.

The start walking and walking is implemented in a function. (Note: The function can be found in /simulations/control/path\_calc.m.)

The path planner will be tested in two different cases; one testing the timer and one testing if positions is asked for in the two walk periods.

The timer shall count from 0 to 0.6[s] with a time steep 0.1[s], and start from 0 again. This is tested by calling the implemented function from a while loop with its time output from last iteration as input. The result of this test is illustrated on Figure 15.3.

This figure illustrates that the timer is working as desired.

The second test shall demonstrate that the path planner is giving the correct position vectors for current phase of the walk, this is triggered of the supervisor by  $F_D$ . The result of the test is shown in Table 15.1.

	$F_D$	
	2	3
$\bar{P}_{P,LF}$	$[0.1462 \quad -0.0717 \quad 0.0152]^T$	$[0.1462 \quad 0.0200 \quad 0.0152]^T$
$\bar{P}_{P,RF}$	$[0.1462 \quad 0.0200 \quad -0.0152]^T$	$[0.1462 \quad -0.0717 \quad -0.0152]^T$
$\bar{P}_{P,LR}$	$[0.1462 \quad 0.0200 \quad 0.0152]^T$	$[0.1462 \quad -0.0717 \quad 0.0152]^T$
$\bar{P}_{P,RR}$	$[0.1462 \quad -0.0717 \quad -0.0152]^T$	$[0.1462 \quad 0.0200 \quad -0.0152]^T$

**TABLE 15.1:** Test of output position vectors from path planner.

When  $F_D$  is 2 are LF and RR in the swing phase and when 3 are RF and LR in the



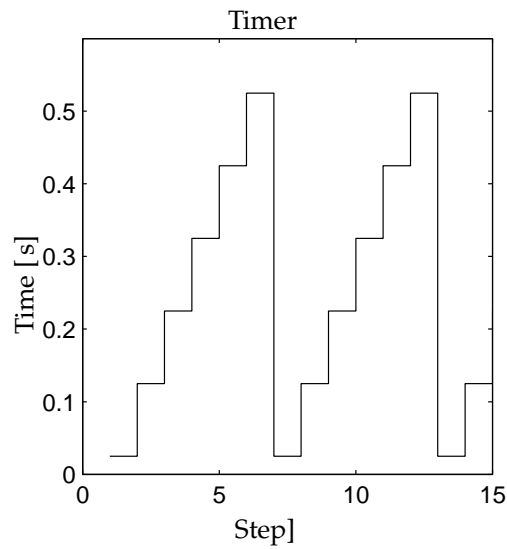
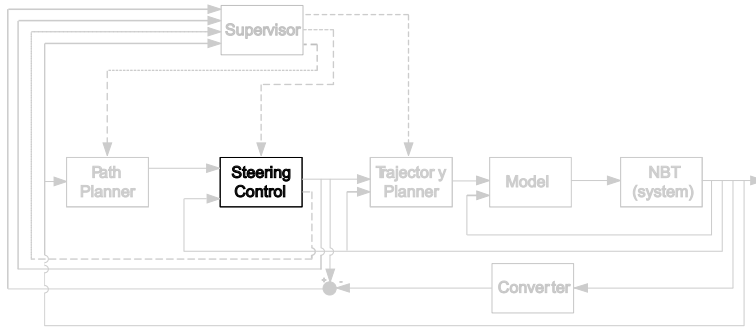


FIGURE 15.3: *An illustration of the timer counting up and resetting.*

swing phase. The position vectors are equal to the position vectors in Equation (15.1) and (15.4), which are as desired.

(Note: The test of the path planner is made in MATLAB<sup>®</sup> and can be found in `/simulations/control/path_planner_test.m`.)

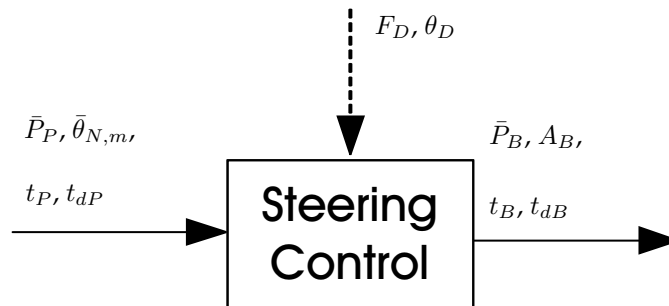




## STEERING CONTROL

*This chapter contains the design of the steering controller. First the controller structures will be discussed and a suited one will be selected. The selected structure will be designed. Afterward the max body angle will be found together with position vectors for this angle. Finally will the system be implemented and tested.*

The claims to the steering controller is that it shall secure that the NBT not falls on the ground by correcting the position vectors for the feet. The inputs and outputs to the steering controller are described in the control system architecture in Chapter 7, the steering controller part is illustrated on Figure 16.1.



**FIGURE 16.1:** An illustration of the steering controller with inputs and outputs. The illustration is a part of the system diagram illustrated on Figure 7.1.

The steering controller will be based on the analysis of quadruped walk made in Chapter 4. This found that the balancing is dynamic and that it can be modeled as an inverted pendulum. An inverted pendulum is unstable, therefore has the steering controller constantly to correct the movements to avoid falling over. The acceleration of the fall depend on the displacement of the center of pressure form the line between the supporting feet. A non linear model of the pendulum is derived in Chapter 13, this is shown again in

---

Equation (16.1).

$$\begin{aligned}
\ddot{\vartheta}_{CM}(i+1) &= \frac{g \cdot \sin(\arctan(\frac{L_{CM}}{H_{CM}}) + \frac{\theta_{B,y}(i)}{\cos(\psi_{IP})} + \frac{\theta_{B,x}(i)}{\sin(\psi_{IP})})}{\sqrt{(L_{CM}^2 + H_{CM}^2)}} \\
\dot{\vartheta}_{CM}(i+1) &= \dot{\vartheta}_{CM}(i) + \ddot{\vartheta}_{CM}(i+1) \cdot t_s \\
\vartheta_{CM}(i+1) &= \vartheta_{CM}(i) + \dot{\vartheta}_{CM}(i) \cdot t_s + \frac{1}{2} \cdot \ddot{\vartheta}_{CM}(i+1) \cdot t_s^2 \\
L_{CM} &= \sin(\psi_{CM} - \psi_{IP}) \sqrt{(\bar{P}_{CM,x} + |\bar{P}_{LR,x}|)^2 + (\bar{P}_{CM,y} + |\bar{P}_{LR,y}|)^2} \\
H_{CM} &= P_{LR,z} + (P_{LR,z} - P_{RF,z}) \\
&\quad \cdot \frac{\tan(\psi_{CM} - \psi_{IP}) \cdot L_{CM}}{\sqrt{(|P_{RF,y}| + |P_{LR,y}|)^2 + (|P_{RF,x}| + |P_{LR,x}|)^2}} \\
\psi_{IP} &= \arctan\left(\frac{|\bar{P}_{RF,x}| + |\bar{P}_{LR,x}|}{|\bar{P}_{RF,y}| + |\bar{P}_{LR,y}|}\right) \\
\psi_{CM} &= \arctan\left(\frac{\bar{P}_{CM,x} + |\bar{P}_{LR,x}|}{\bar{P}_{CM,y} + |\bar{P}_{LR,y}|}\right)
\end{aligned} \tag{16.1}$$

The steering control can be made with different purposes, two of interest are to minimize the rotations of the body or use the fall to make the forward motion, these will be outlined in the following.

The steering control has nine inputs and seven outputs which makes it a MIMO system, therefore an obvious choice for control to minimize the rotations of the body is state-space. State-space control easily can be used for MIMO systems and bases it's control on estimations of the system states.

The unstableness of the inverted pendulum construction makes the NBT fall, this can be used to make the forward motion of the body. The legs in the swing phase then have to be put down on the floor before the body angle gets to large, thereby catching the body every time the swinging legs hit the floor, making the body rotate in a limited circle.

The rotations of the body are important to minimize because this is a criteria for trot locomotion, but it is also important to make the locomotion fast while securing that the NBT not is falling as the problem formulation states. By letting the body fall forward and catching the fall with the legs in the swing phase, the fall helps increasing the velocity of the walk and falling still is avoided. The fall produces a rotation that the swinging legs shall counter act on when they are put on the floor, this is discussed in section 4.1. The decided steering controller get the limitation that it first can correct on a fall when the feet is set on the floor in the end of the phase, because the walk shall be kept synchronous. Therefore the control purpose have been chosen to let the body fall forward and catch the fall when the legs in the swing phase are put on the ground. The position vectors for where the feet shall be in the end of the swing phase  $\bar{P}_B$  depends on

the angle of the body  $\bar{\theta}_{N,m}$  and the input position vectors  $\bar{P}_P$ . The controller is chosen to be a ratio controller calculating a ratio for how much  $\bar{P}_P$  shall be changed against a maximum position vector  $\bar{P}_M$  from the angle of the body. The control system will be connected as illustrated on Figure 16.2.

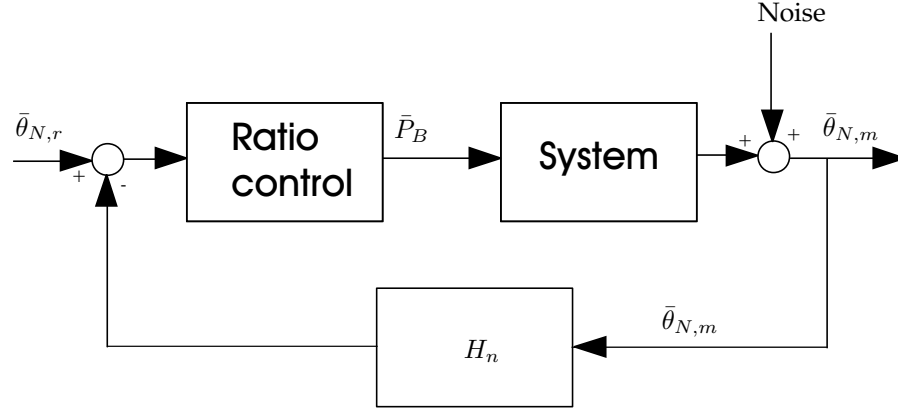


FIGURE 16.2: An illustration of the steering controller system with feed back signal.

The system block contains the trajectory planner, inverse manipulator kinematics model, dynamics model and the NBT (system). The input to the ratio controller is the error between the reference  $\bar{\theta}_{N,r}$  and the measured output  $\bar{\theta}_{N,m}$ . The angle of the body  $\bar{\theta}_{N,m}$  is a feed back signal, where some random white noise is expected. The noise can have different origins, two of the likely are sensor noise and the resolution of the motors.  $\bar{\theta}_{N,r}$  is the reference angle for the body, which in trot is zero for forward motion.  $H_n$  is the feedback gain, which will be set to 1, so the measured body angle is feedback with out gain.

The ratio can be made with different controller types, of interest are Proportional P, Proportional Integrating PI and Proportional Integrating Derivative PID, these will be discussed in the following.

The P controller will change the ratio linearly, with out considering the offset errors and noise. The I part of the controller will further integrate the error and over time eliminate it, the integrates can however give overshoot which is undesirable. The D part of the controller will derivate the input and react fast on changes in the error, it will however give an rapidly change of the output if the input changes as a step which also is undesirable. When the legs are shifted is the direction of the fall also changed, then the integrator will give an overshoot, it is therefore rejected as a solution. If the NBT gets in contact with an object the error will change rapidly and the derivative will consequently also make the output change rapidly which is undesirable. Therefore will the ratio be made with a P controller.

---

The gain in the P controller shall give the ratio  $R_B$  for how much  $\bar{P}_P$  shall be corrected toward  $\bar{P}_M$  with the current input error. Since  $\theta_{N,r}$  is zero in trot can it be disregarded in the calculation. The formula for calculating  $R_B$  is derived in Equation (16.2), where  $k$  is the gain in the P controller.

$$\begin{aligned} R_B &= k \cdot (\theta_{N,r} - \theta_{N,m}) \\ &= \frac{1}{\theta_D - \theta_{N,r}} \cdot (\theta_{N,r} - \theta_{N,m}) \\ &= \frac{-\theta_{N,m}}{\theta_D} \end{aligned} \quad (16.2)$$

The correction in the x, y and z directions of the coordinate system  $\{LF0\}$ ,  $\{RF0\}$ ,  $\{LR0\}$  and  $\{RR0\}$  are derived in Equation (16.3).

$$x = (P_{M,x} - P_{P,x}) \cdot R_B \quad (16.3a)$$

$$y = (P_{M,y} - P_{P,y}) \cdot R_B \quad (16.3b)$$

$$z = (P_{M,z} - P_{P,z}) \cdot R_B \quad (16.3c)$$

These corrections are added to the position vectors from the path planner in Equation (16.4).

$$P_{B,x} = P_{P,x} + x \quad (16.4a)$$

$$P_{B,y} = P_{P,y} + y \quad (16.4b)$$

$$P_{B,z} = P_{P,z} + z \quad (16.4c)$$

$\bar{P}_M$  depends on the working range of the legs in sideways direction. The maximum angle the legs can be moved sideways are to the boundary of the working range, which depend on the leg as described in Chapter 10. The boundary for the rotation in under the body is  $\pi/4$ [rad] and away from the body  $\pi/2$ [rad]. The limit is in the side opposite the fall, therefore will this be used to find  $\theta_D$ .  $\bar{P}_M$  for the rotation in under the body with the other two joint angles as defined for the swing phase in Chapter 15 is calculated in Equation (16.5). These position vectors are between the coordinate system locked to the shoulders and the foot of the current leg.

$$\bar{P}_{M,opp,LF} = [0.1082 \quad -0.0450 \quad -0.1030]^T \quad (16.5a)$$

$$\bar{P}_{M,opp,RF} = [0.1082 \quad -0.0450 \quad 0.1030]^T \quad (16.5b)$$

$$\bar{P}_{M,opp,LR} = [0.1082 \quad -0.0450 \quad 0.1030]^T \quad (16.5c)$$

$$\bar{P}_{M,opp,RR} = [0.1082 \quad -0.0450 \quad -0.1030]^T \quad (16.5d)$$

When the NBT is falling the shoulder of the leg in the opposite side of the fall will be lifted and therefore the distance to the floor increased as illustrated on Figure 16.3.

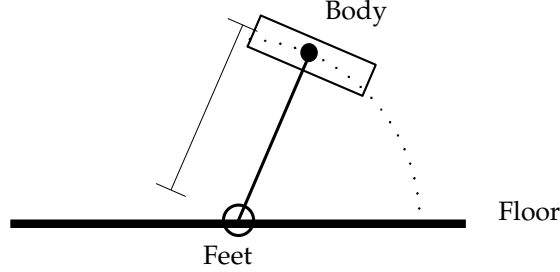


FIGURE 16.3: An illustration of the height change from the shoulder to the floor when falling.

Then can the foot not reach the floor, therefore has the height of the body to be lowered, this can be done when the problem occurs or it can be lowered for the whole walk sequence. If the height is lowered when stopping the fall, there will be generated some forces and torques by the height change, one of the extra forces is the gravity of the free fall when changing height which make the inverted pendulum fall faster. It is undesirable to introduce extra dynamics, therefore the height for the whole walk will be lowered. The x component of the position vectors in Equation (15.1) and (15.2) will be changed. The knee can bend  $\pi/2$ [rad], this makes the minimum length the leg can be bended up under the body as in Equation (16.6).

$$\begin{aligned} L_{min} &= \sqrt{((L_{LF2} + L_{LF3})^2 + L_{LF4}^2)} \\ &= \sqrt{((0.0110 + 0.0656)^2 + 0.0866^2 + 0.0110^2)} \\ &= 0.1161[\text{m}] \end{aligned} \quad (16.6)$$

The lengths are from the derived model for the manipulator kinematics and can be found in Appendix A.  $L_{min}$  is the minimum length the leg can get when standing on it. This length is larger then the x component of the position vectors in Equation (16.5), therefore have these position vectors to be changed. The rotation of  $\theta_{LF2}$ ,  $\theta_{RF2}$ ,  $\theta_{LR2}$  and  $\theta_{RR2}$  will be changed to  $\pm\pi/5$ [rad] and the rotation of  $\theta_{LF1}$ ,  $\theta_{RF1}$ ,  $\theta_{LR1}$  and  $\theta_{RR1}$  will be changed to  $-\pi/7$ [rad], this gives the position vectors in Equation (16.7).

$$\bar{P}_{M,opp,LF} = [0.1256 \quad -0.0450 \quad -0.0895]^T \quad (16.7a)$$

$$\bar{P}_{M,opp,RF} = [0.1256 \quad -0.0450 \quad 0.0895]^T \quad (16.7b)$$

$$\bar{P}_{M,opp,LR} = [0.1256 \quad -0.0450 \quad 0.0895]^T \quad (16.7c)$$

$$\bar{P}_{M,opp,RR} = [0.1256 \quad -0.0450 \quad -0.0895]^T \quad (16.7d)$$

---

The height of the body doing ordinary walk is changed to 0.1180[m], which is a little more than  $L_{min}$ , but in the swing phase the foot is lifted 0.01[m] from the floor by the trajectory planner. This lift is at the middle of the swing curve where the feet are out in  $-0.0259$ [m] therefore is the chosen height expected to be achievable. The position vectors in Equation (15.1) and (15.2) are changed to the position vectors in Equation (16.8) and (16.9), respectively.

$$\bar{P}_{P,LF} = [0.1180 \quad -0.0717 \quad 0.0152]^T \quad (16.8a)$$

$$\bar{P}_{P,RF} = [0.1180 \quad -0.0717 \quad -0.0152]^T \quad (16.8b)$$

$$\bar{P}_{P,LR} = [0.1180 \quad -0.0717 \quad 0.0152]^T \quad (16.8c)$$

$$\bar{P}_{P,RR} = [0.1180 \quad -0.0717 \quad -0.0152]^T \quad (16.8d)$$

$$\bar{P}_{P,LF} = [0.1180 \quad 0.0200 \quad 0.0152]^T \quad (16.9a)$$

$$\bar{P}_{P,RF} = [0.1180 \quad 0.0200 \quad -0.0152]^T \quad (16.9b)$$

$$\bar{P}_{P,LR} = [0.1180 \quad 0.0200 \quad 0.0152]^T \quad (16.9c)$$

$$\bar{P}_{P,RR} = [0.1180 \quad 0.0200 \quad -0.0152]^T \quad (16.9d)$$

The calculation of  $\theta_D$  will be made as an example where the LF leg will be moved in under the body, then RF and LR will be in the stand phase.  $\theta_D$  will be calculated when the state is shifted because all the feet then are on the ground, therefore are the position of the LF, RF and LR foot known as illustrated on Figure 16.4(a).  $L_{LF}$  is the length from the line of support to the LF foot. The position vectors in the example will be from coordinate system  $\{1\}$  to the feet instead of from the coordinate systems locked to the shoulders, because this will make the equations in the example more transparent. The placement of coordinate system  $\{1\}$  is illustrated on Figure 16.4(b).

$L_{LF}$  will be found by making some RAT as illustrated on Figure 16.5(a) and 16.5(b).

The length of  $L_{LF}$  is calculated in Equation (16.10).

$$\begin{aligned} \psi_{IP} &= \arctan\left(\frac{|P_{RF,x}| + |P_{LR,x}|}{|P_{RF,y}| + |P_{LR,y}|}\right) \\ \psi_{LF} &= \arctan\left(\frac{|P_{LF,x}| + |P_{LR,x}|}{|P_{LF,y}| + |P_{LR,y}|}\right) \\ L_{LF} &= \sin(\psi_{LF} - \psi_{IP}) \sqrt{((P_{LF,x} + |P_{LR,x}|)^2 + (P_{LF,y} + |P_{LR,y}|)^2)} \quad (16.10) \end{aligned}$$



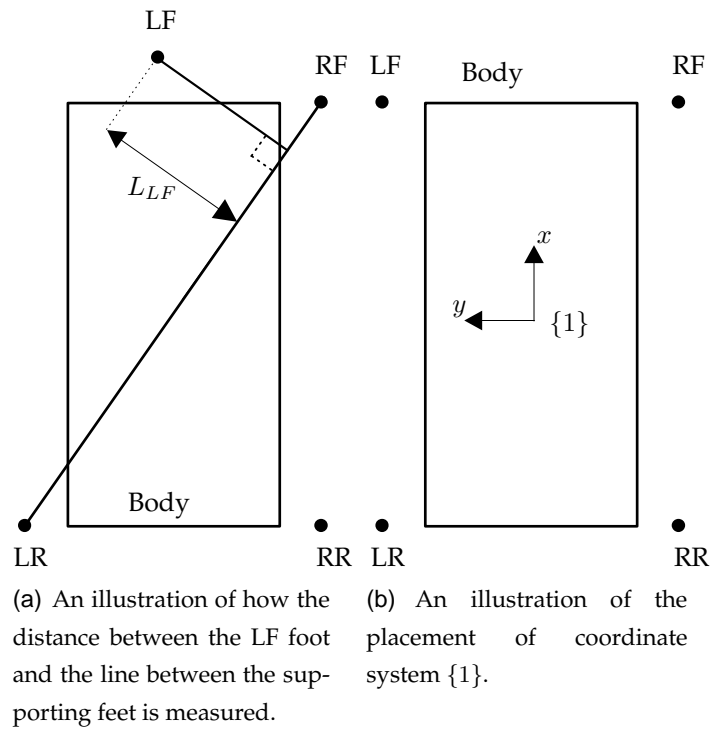


FIGURE 16.4: An illustrations of the NBT seen for above.

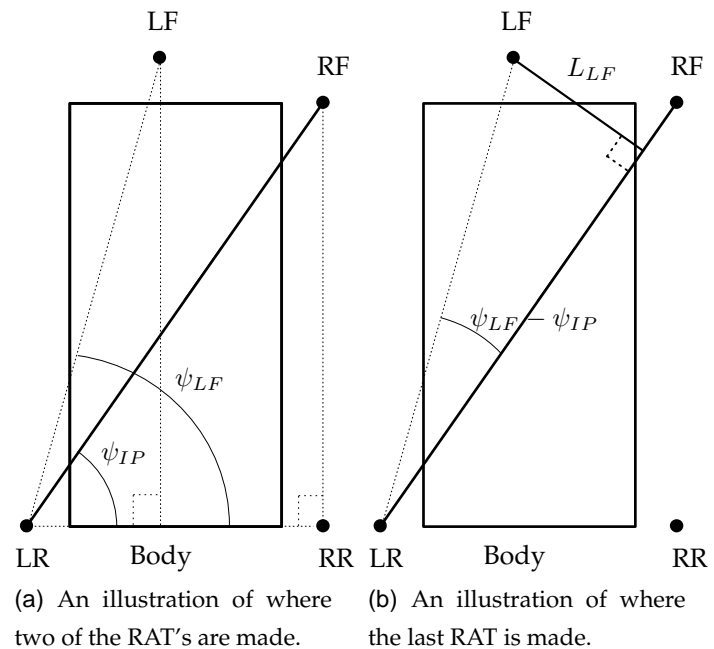


FIGURE 16.5: Illustrations of how the RAT is made to calculate  $L_{LF}$ .

$L_{LF}$  can be used to make a RAT where  $\theta_{D1}$  can be found, as illustrated on Figure 16.6. The height from the floor to the LR and RF shoulder are similar in the walk cycle, therefore the shoulder heights of the rear supporting leg is used.

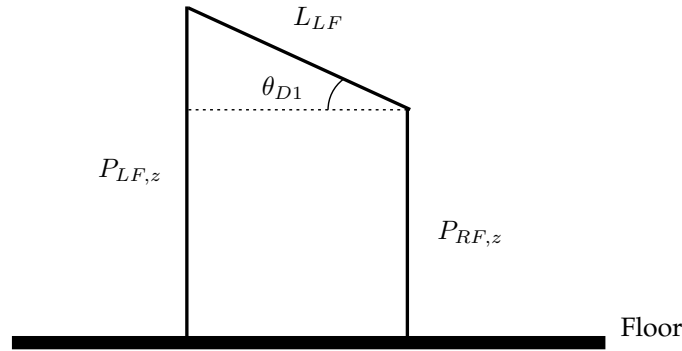


FIGURE 16.6: An illustration of the RAT made to calculate  $\theta_{D1}$ .

The angle  $\theta_{D1}$  is calculated in Equation (16.11a) and changed to  $\theta_D$  in Equation (16.11b).

$$\theta_{D1} = \pm \arcsin \left( \frac{L_{LF}}{\frac{|P_{RF,z}| + |P_{LR,z}|}{2}} \right) \quad (16.11a)$$

$$= \pm 0.3667 [\text{rad}]$$

$$\theta_D = \pm \theta_{D1} \cdot \sin(\psi_{IP}) \quad (16.11b)$$

$$= \pm 0.3234 [\text{rad}]$$

$\bar{P}_M$  for the side that the fall is against can now be calculated, this will be different for the front- and rear legs because the distance between the shoulders are different, as illustrated on Figure 16.7, the distances are used in the deriving of the kinematics model, these can be found in Appendix A.

The position vectors in Equation (16.5) will be used to calculate the other positions vectors. The x component of the position vector is calculated in Equation (16.12). The position vectors for the feet shall as defined in Chapter 10 be given in relation to the

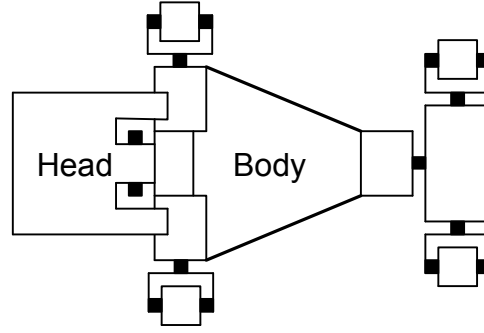


FIGURE 16.7: The NBT seen from above, standing on its legs with the head up.

coordinate systems locked to the shoulders, therefore will this be used in the sequel.

$$\begin{aligned}\bar{P}_{M,sid,LF,x} &= \bar{P}_{M,opp,RF,x} - 2 \cdot d_{LF1} \cdot \tan(\theta_D) \\ &= 0.0891[\text{m}]\end{aligned}\quad (16.12a)$$

$$\begin{aligned}\bar{P}_{M,sid,RF,x} &= \bar{P}_{M,opp,LF,x} - 2 \cdot d_{LF1} \cdot \tan(\theta_D) \\ &= 0.0891[\text{m}]\end{aligned}\quad (16.12b)$$

$$\begin{aligned}\bar{P}_{M,sid,LR,x} &= \bar{P}_{M,opp,RF,x} - 2 \cdot L_{RR1a} \cdot \tan(\theta_D) \\ &= 0.1008[\text{m}]\end{aligned}\quad (16.12c)$$

$$\begin{aligned}\bar{P}_{M,sid,RR,x} &= \bar{P}_{M,opp,RF,x} - 2 \cdot L_{RR1a} \cdot \tan(\theta_D) \\ &= 0.1008[\text{m}]\end{aligned}\quad (16.12d)$$

$d_{LF1}$  and  $L_{RR1a}$  are half of the distance between shoulders in the front and rear, respectively. These variables are used in the *Denavit-Hartenberg notation* which were used deriving the manipulator kinematics model.

The y component of the position vectors are the distance the foot is moved in forward and backward direction, this will be kept similar to the position vectors in the opposite side of the fall direction because both the legs in the swing phase then moves the feet the same length forward. With the same argument will the z component also be chosen to the same, then will both the corrected legs move the same length in the stand phase. This is giving the Position vectors in Equation (16.13).

$$\bar{P}_{M,sid,LF} = [0.0891 \quad -0.0450 \quad 0.1030]^T \quad (16.13a)$$

$$\bar{P}_{M,sid,RF} = [0.0891 \quad -0.0450 \quad -0.1030]^T \quad (16.13b)$$

$$\bar{P}_{M,sid,LR} = [0.1008 \quad -0.0450 \quad -0.1030]^T \quad (16.13c)$$

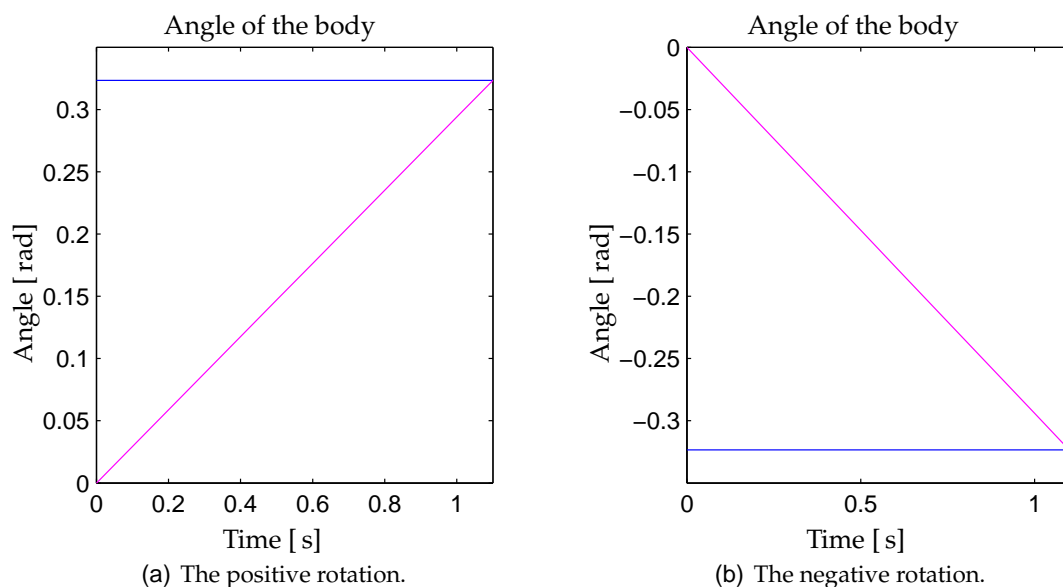
$$\bar{P}_{M,sid,RR} = [0.1008 \quad -0.0450 \quad 0.1030]^T \quad (16.13d)$$

The maximum period time doing walk where found to 0.6[s] in Chapter 15, it has to be checked if the body angle is getting larger then  $\theta_D$  in 0.6[s], because the rotations of the body then will be outside the limited cycle and the system will be unstable, the steering controller then wont be able to stop the fall. The period time will be tested when the system is combined.

The stability of the system together with the controller can be tested with different principals. One will be to linearize the model of the inverted pendulum, e.g. by use of small signal approximations, and use the Root-Locus design method to secure that the limited circle is stable [GFFP02, Chapter 5.7]. Another method is to use the Lyapunov stability criteria, which can be used to evaluate if the controller together with the nonlinear system is stable [GFFP02, Chapter 3]. The test of the stability is not the scope of this project, therefore will it be disregarded.

The steering controller is implemented as a function calculating  $\bar{P}_B$  for the legs. (Note: The function can be found in /simulations/control/steering\_control.m.)

The implementation is tested by letting it calculate the corrected position vector  $\bar{P}_B$  for different body angles  $\theta_{N,m}$ . The body angles will start as zero and increase linearly until  $\theta_{N,m}$  is equal to  $\theta_D$  in one test and in another test decrease until equal to  $-\theta_D$ , as illustrated on Figure 16.8.



**FIGURE 16.8:** An illustration of the body angles used in the test. The blue line is for  $\theta_D$  and the magenta line is for  $\theta_{N,m}$ .

The input vector will be the ones in Equation (16.8) for the swing phase and the ones in Equation (16.9) for the stand phase. The test are similar for all the legs therefore will only the test for the LF leg be documented, the results for the other tests can be found in `/simulations/control/steering_control_test.m`

The test will be performed for falls in both directions which is measured around the x axis of coordinate system {1}. The result of the test with fall in positive direction is illustrated on Figure 16.9(a) and in negative direction on Figure 16.9(b).

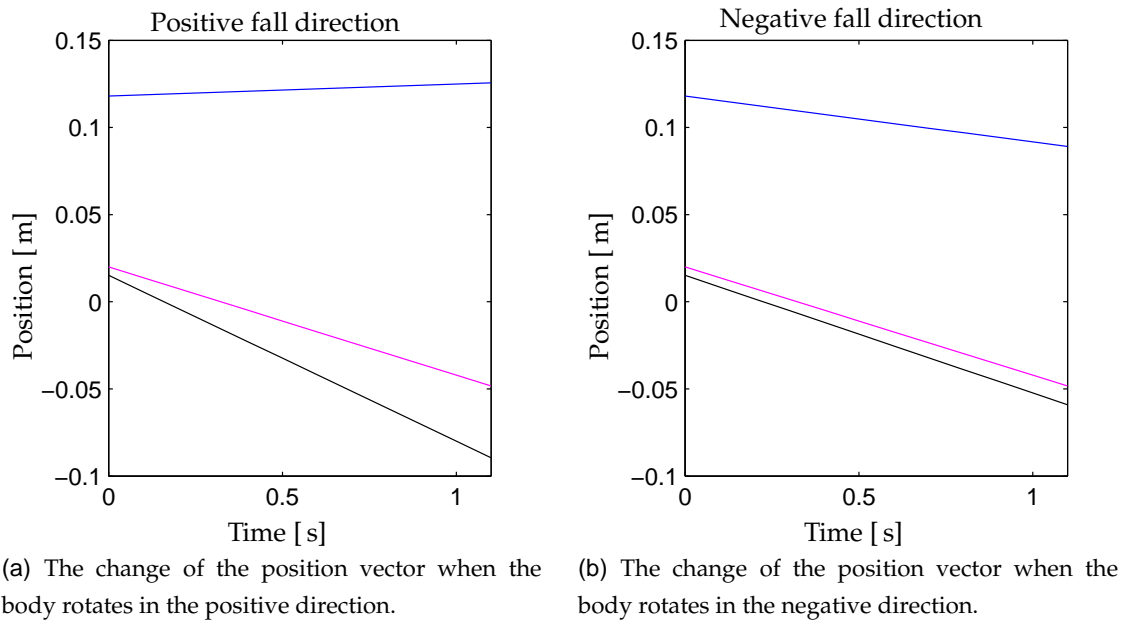
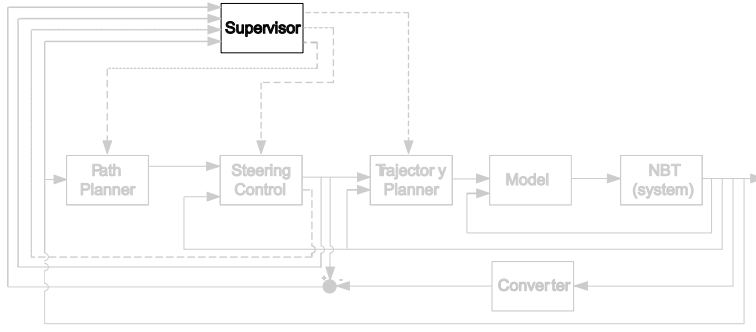


FIGURE 16.9: Test results for the LF leg. The blue line is for x, the magenta line is for y and the black line is for z direction.

The figures demonstrates that the steering controller changes the three components of the position vectors for where the foot shall be moved linearly toward their maximum position vector, which are shown in Equation (16.7a) and (16.13a) for positive and negative rotation of the body, respectively. The steering controller will be tested more dynamic in the system test, but this test demonstrated that the corrections were in the right directions.





## SUPERVISOR

*This chapter contains the design of the supervisor. First is the structure introduced, afterwards is the system limited and implemented..*

The supervisor generate the time discrete events that keeps the parts in the control system synchronous, as for the path planner the supervisor will be kept as simple as possibly, because there always is a more sophisticated one for the NBT, this one therefore not will be used. This means that there only will be designed a supervisor for the walking cycle and the start walk will be disregarded. The supervisor controls which leg there are in the swing and stand phase, when the walk is started, the desired velocity and the max acceptable rotation of the body before the steering control shall stop the walk.

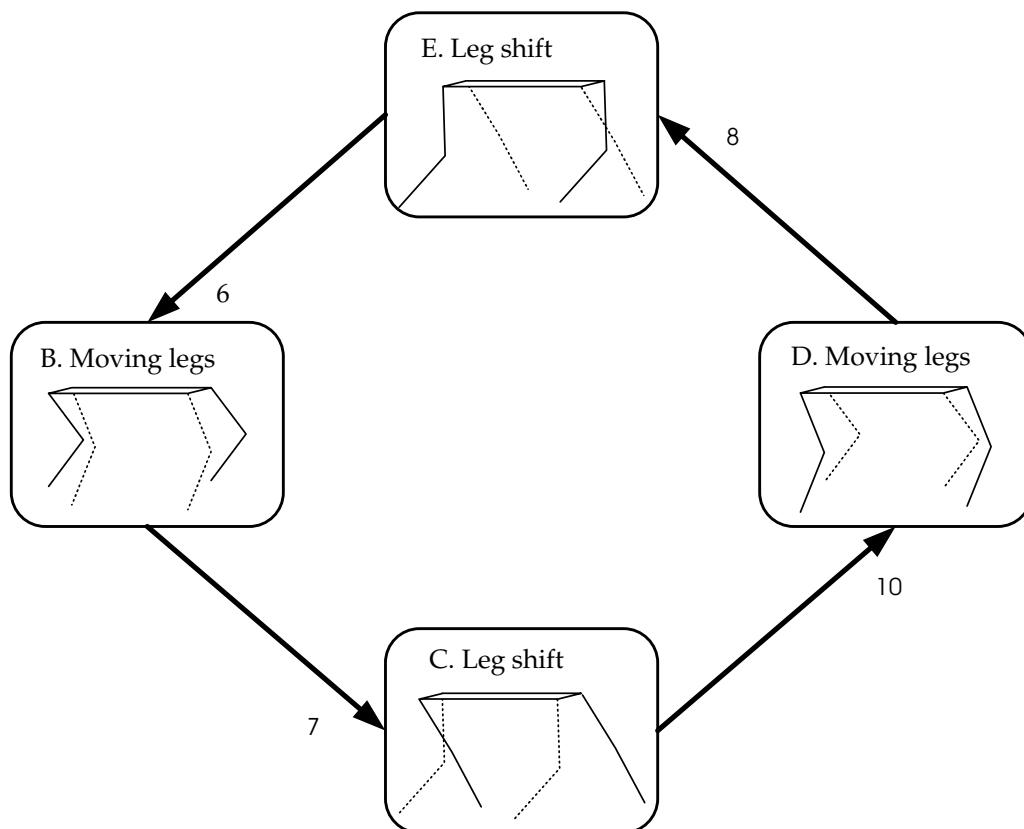
The inputs and outputs to the path planner are described in Chapter 7, the supervisor is illustrated in a rearranged form on Figure 17.1.



**FIGURE 17.1:** An illustration of the supervisor with inputs and outputs. The illustration is a part of the system diagram illustrated on Figure 7.1.

The construction of the supervisor will be based on the state machine analyzed in Chapter 5. The four walking states from Figure 5.3 are illustrated on Figure 17.2.

In state B and D are one LP in the swing phase and the other are in the stand phase. When  $t_{dP} - t_s$  is equal to  $t_P$  is the state shifted to state C or E as Figure 17.2 illustrated. State C and E in the state machine are securing that all the feet are on the ground when the legs in the swing and stand phase are shifted, this can be secured with different methods, of interest are; comparing of the desired and obtained position of the feet or load changes in the motors. Comparing of the positions secure that the desired position are obtained,



**FIGURE 17.2:** *An illustration of the walking cycle in the state machine.*

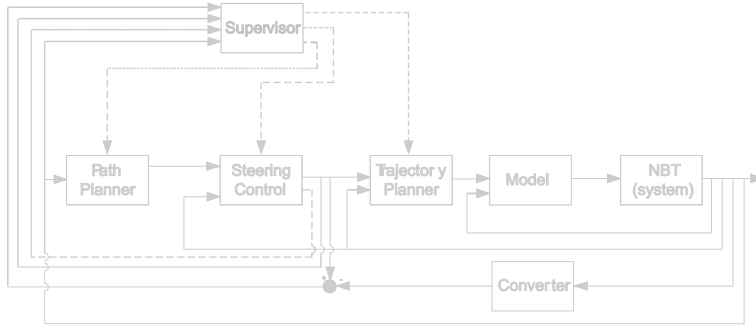


but not whether the feet are on the ground or not. When a foot make impact with the ground the load sensor in the motors measure a radical increase in load, this also happen if the leg hid something or the foot touch the ground doing the swing phase. Both the methods can misinterpret the informations and this is not acceptable, therefore has to be considered other solutions. One solution is to combine the two methods because they have different conditions for misinterpreting the information. This solution will be advantageous to implement on the real NBT, but in the simulations will the radical change in the load sensor signal come when the foot have obtained the desired position, because the load sensor signal would be made partly from if the desired position is obtained. Therefore if the position is obtained, the state will be changed as Figure 17.2 demonstrates. Noise in the sensors and the resolution of the motors will give a deviation between the obtained and desired position, therefore an acceptable margin of error  $E_d$  has to be found. The scale of the noise in the sensors are unknown and because the prototype of the NBT not is at disposal, can the noise in the sensors not be tested. Therefore the resolution of the motors will be used to calculate margin with. The motors have a resolution of  $0.0061[\text{rad}]$  as described in Chapter 3. The margin will be calculated with straight leg and the shoulder joint rotated  $0.0061[\text{rad}]$  in Equation (17.1).

$$\begin{aligned} E_d &= \tan(0.0061) \cdot (0.0866 + 0.0656 + 0.011) \\ &= 0.001 [\text{m}] \end{aligned} \tag{17.1}$$

(Note: The implementation of the supervisor can be found  
in `/simulations/control/supervisor.m`.)





## SYSTEM TEST

*This chapter concerns the test of the controller system. The tests in this chapter will show if the control system is performing as expected and which limitations it has. First the system will be combined afterward it will be tested and finally the results will be discussed.*

The control system parts will be combined as outlined in Chapter 7 with the outputs from the *NBT (System)* calculated with the model parts as they were combined in Chapter 12. The feedback of  $P_m$  to the *trajectory planner* is made. The Movements of the NBT will be visualized in the game engine modified in Chapter 6. All the control system parts are made in MATLAB<sup>®</sup> and Simulink<sup>®</sup>, therefore the control system is implemented in Simulink<sup>®</sup>. The Simulink<sup>®</sup> implementation is illustrated as a block diagram on Figure 18.1.

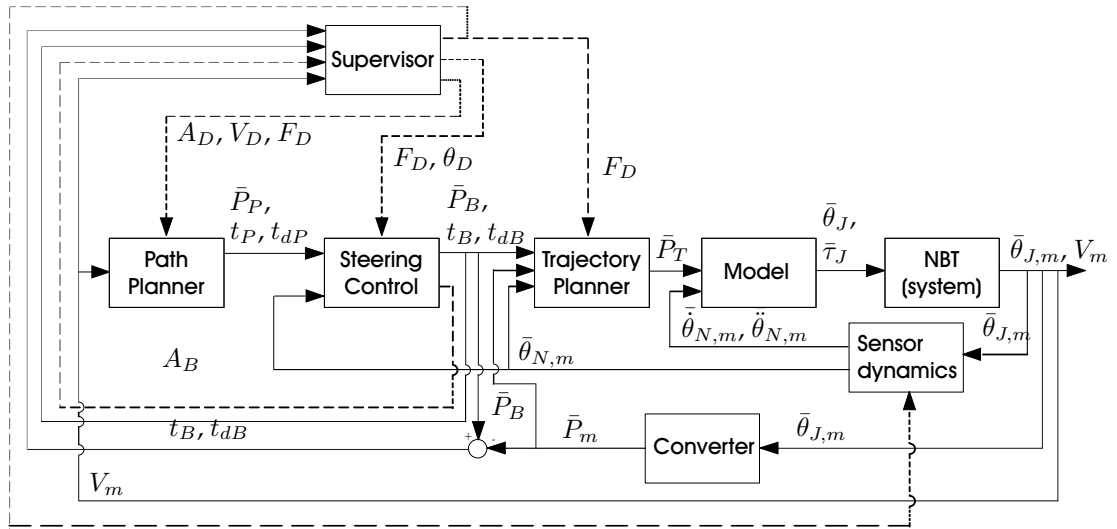


FIGURE 18.1: An illustration of the implementation in Simulink<sup>®</sup>.

The *Sensor dynamics* block includes the rotational dynamics made in Chapter 13 and the function to calculate the location of the CM for the entire NBT made in Chapter 11. The function for the CM is also used in the dynamics model to divide the pressure be-

tween the legs, here is the function calculation the location of the CM from the calculated joint angles and in the sensor is it from the obtained joint angles. The NBT will be walking on the spot in the visualization, because a sensor for the movement of the body is not implemented. The implementations of the control parts are with a samplings frequency of 10 [Hz], therefore the combined system will also get this. (Note: The combining of the system can be found in `/simulation/control/combined.mdl`.)

The combined control system will be tested in three test cases; first the period time will be tested and a suited one will be selected, second the stability of the combined system and the calculated joint angles will be tested, the third test is similar to test two with white noise on the feedback signals. The second test will demonstrate that all the parts are working as intended and the third will demonstrate how robust the control system is against disturbances. The tests are performed in the three following sections. The dotted vertical lines on the figures illustrate the phase shift. (Note: The illustrations and calculations in this chapter can be calculated with `/simulation/control/system_test.m`.)

## 18.1 PERIOD TIME

The period time will be tested to find a suited one for the walk, where the steering controller can keep the angle of the body inside the limited circle with  $\theta_D$  as boundary. The maximum period time for the walk was found to 0.6 [s] in Chapter 15 to obtain the minimum velocity of the walk. The new period time is found by simulating one period of the walk and find the point where  $\vartheta_{CM}$  becomes larger than  $\theta_D$ . The result of the simulation is illustrated on Figure 18.2. The blue line represents rotation around the x-axis, the magenta line, rotation around the y-axis and the black line, rotation around the z-axis of coordinate system  $\{1\}$ .

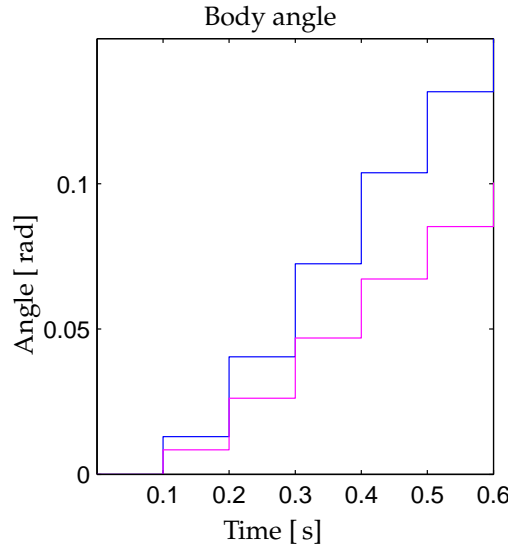
$\bar{\theta}_{N,m,x}$  is increasing to approximately 0.13 [rad] at 0.6 [s],  $\theta_D$  is calculated to  $\pm 0.3234$  [rad] in Chapter 16.  $t_{dP}$  is not changed because  $\bar{\theta}_{N,m}$  is less than half of  $\theta_D$ .

The obtain velocity with this period time is calculated in Equation (18.1).

$$\begin{aligned} V_N &= \frac{0.917 \text{ [m]}}{0.6 \text{ [s]}} \\ &= 1.5283 \text{ [m/s]} \end{aligned} \tag{18.1}$$

The distance is along the floor from the position vectors in Equation (16.8) and (16.9) for ordinary walk. The calculated velocity seams as a reasonable velocity for a running animal.

With a sampling frequency of 10 [Hz] there will be six samples in the stand and swing



**FIGURE 18.2:** An illustration of the how the body angle is increasing over 0.6 [s] due to the fall of the inverted pendulum. The blue line represents rotation around the x-axis, the magenta line, rotation around the y-axis and the black line, rotation around the z-axis of coordinate system {1}.

phase. The accuracy of the sensor to calculate the  $\bar{\vartheta}_{CM}$ ,  $\dot{\bar{\vartheta}}_{CM}$  and  $\ddot{\bar{\vartheta}}_{CM}$  depends on the sampling frequency as discussed in Chapter 13, to increase the accuracy the sampling frequency has to be increased. The visualization shifts linear between the samples, which make the movements in the demonstration awkward only with six samples. The movements of the NBT are also difficult to follow in the visualization during a time period of 0.6 [s], therefore the simulate will be conducted in slow motion. The slow motion time has been chosen to  $1/4$  of real time, because the movements then are possible to study in the visualization. This means that the period times in the simulations and visualizations will be 2.4 [s] and the sampling frequency will be 40 [Hz].

## 18.2 STABILITY

First the phase between the legs will be tested by simulating the walk and examine how  $F_D$  changes, this is illustrated on Figure 18.3(a).  $F_D$  changes when the period ends and the difference between  $\bar{P}_B$  and  $\bar{P}_m$  is smaller than 0.001 [m], as described in Chapter 17. The changes of the x, y and z component of  $\bar{P}_B$  and  $\bar{P}_m$  are illustrated on Figure 18.3(b), 18.3(c) and 18.3(d), respectively, for the RF leg. There is a unit delay of 0.025 [s] between  $\bar{P}_B$  and the feedback signal  $\bar{P}_m$  which is shown in the previously mentioned figures.

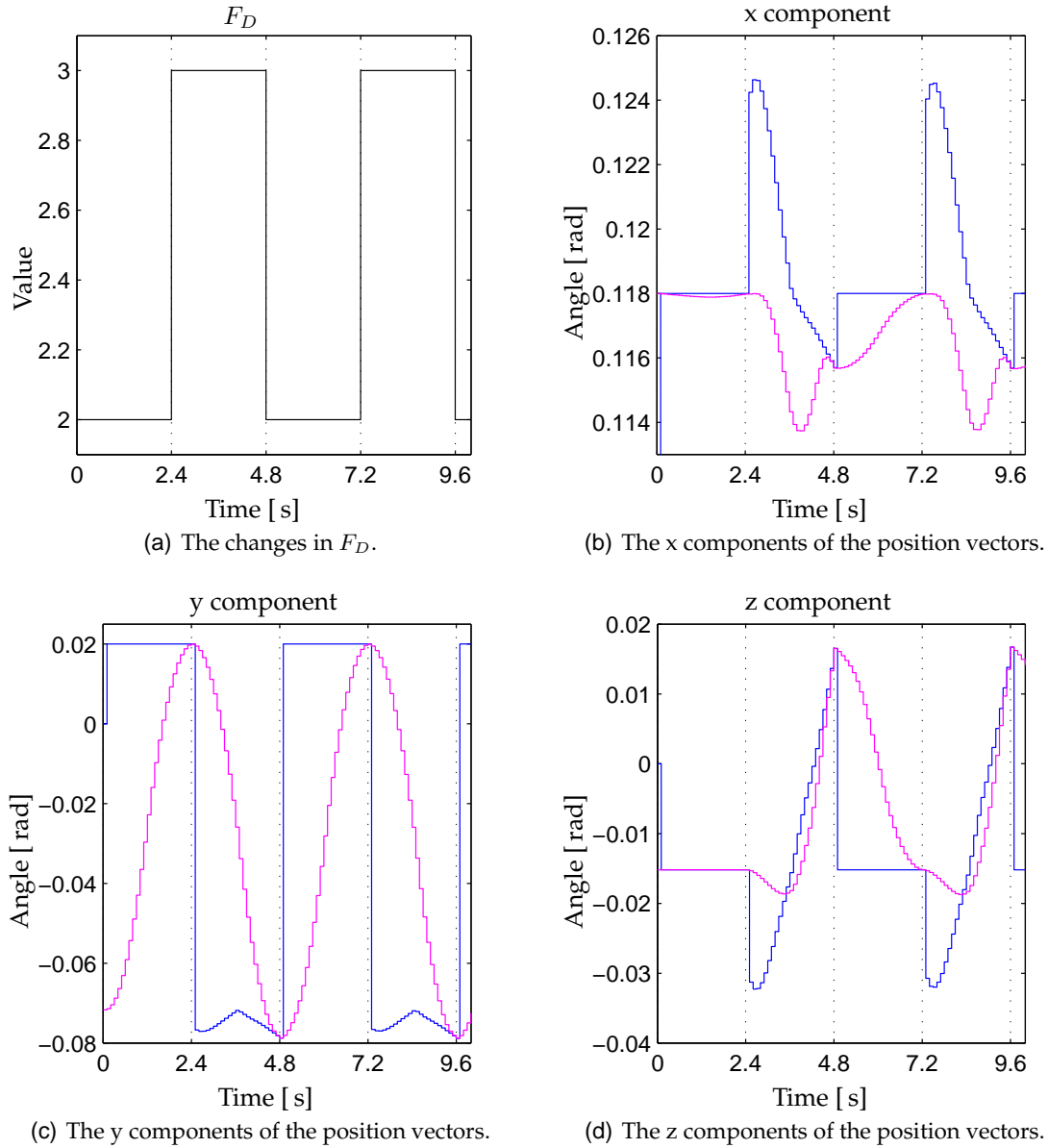


FIGURE 18.3: Test results of walking to examine the phase between the legs.

The blue line represents the component of  $\bar{P}_B$ , the magenta line the component of  $\bar{P}_m$  and the black line  $F_D$ .

$F_D$  is changing between 2 and 3 with an interval of 2.4[s] which is four times the period time of 0.6[s] for each phase of the walk, as earlier mentioned the simulation is in slow-motion with  $1/4$  of real time, therefore the period time is four times larger. The supervisor changes  $F_D$  when the period time is used and the positions of the feet are obtained. The earlier mentioned figures demonstrates that  $\bar{P}_m$  is changing toward  $\bar{P}_B$  and becomes akin to it at the end of each period.  $\bar{P}_B$  is constant in the periods where the

current leg is in the stand phase, and is changing in the direction of the falls in the swing phase. The RF leg is swinging from 2.4 to 4.8[s] and 7.2 to 9.6[s], where the blue line is varying and else is constant. The tendencies of  $\bar{P}_B$  is similar for all the legs, the same goes for the RF, LR and RR leg for  $\bar{P}_m$ , but the LF leg has an error, this is illustrated on Figure 18.4.

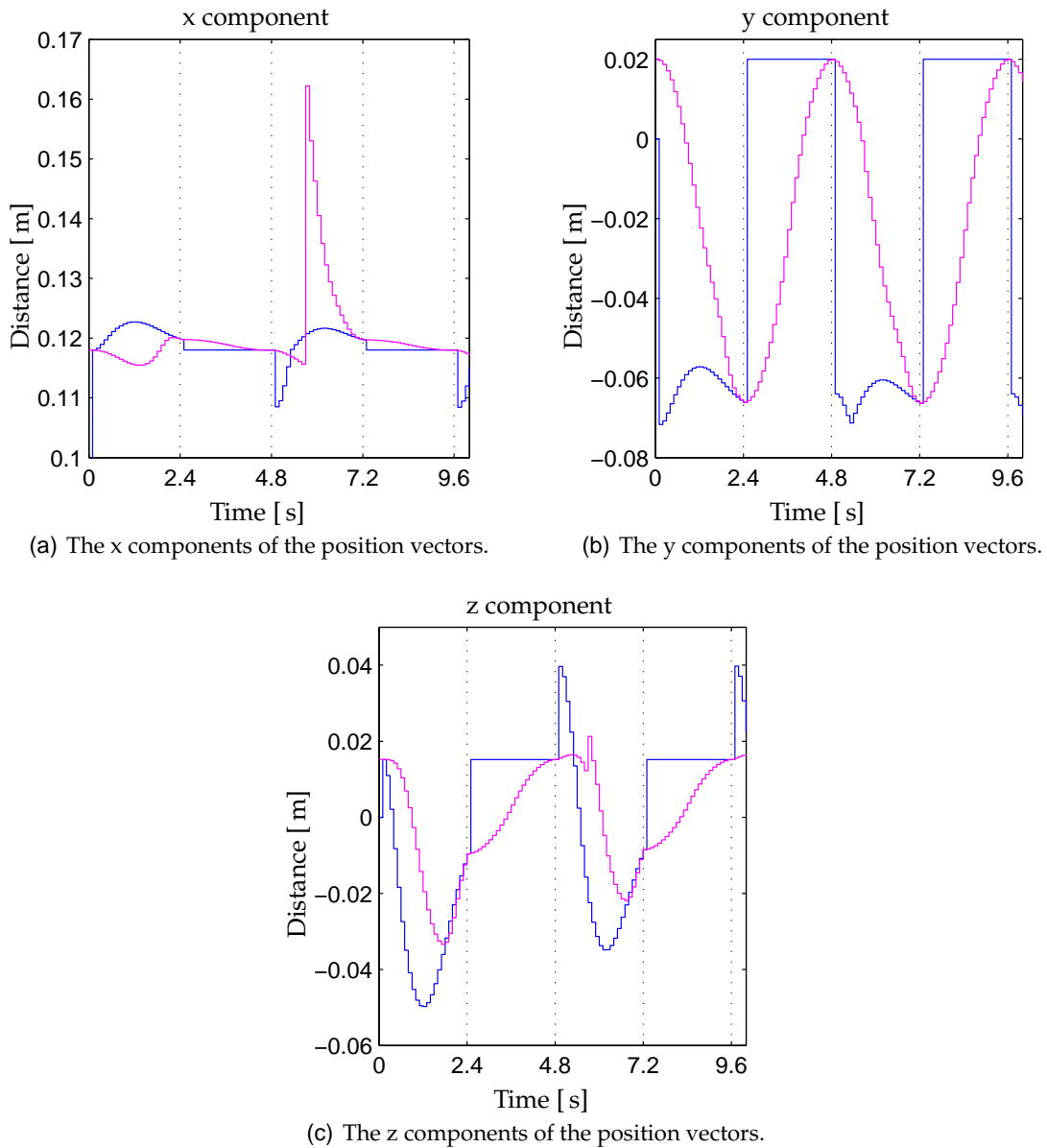
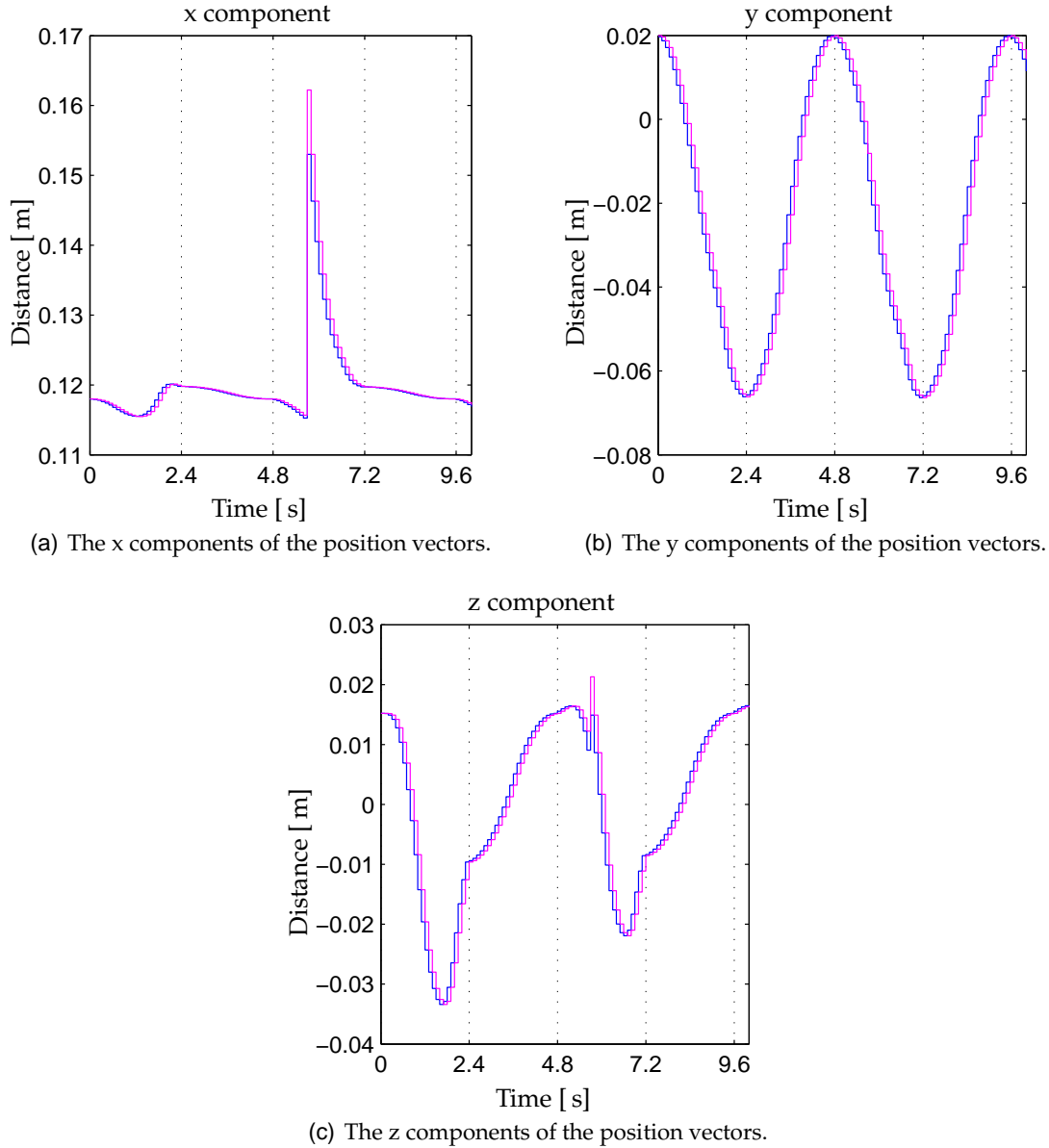


FIGURE 18.4: Test results for the LF leg in walk. The blue line represents the component of  $\bar{P}_B$  and the magenta line the component of  $\bar{P}_m$ .

The x and z component of  $\bar{P}_m$  are making a spike at 5.7[s] of approximately 0.05[m] and 0.01[m], respectively. After the spikes the x and z components are converging against

$\bar{P}_B$ , this is because the trajectory planner is correcting the position of the feet toward the desired position in the end of the period. The position vector from the trajectory planner,  $P_T$ , for the LF leg is illustrated on Figure 18.5. The blue line represents  $\bar{P}_T$  and the magenta line  $\bar{P}_m$ .



**FIGURE 18.5:** Test results for the trajectory planner for the LF leg. The blue line represents the component of  $\bar{P}_T$  and the magenta line the component of  $\bar{P}_m$ .

The figure demonstrates that the trajectory planner is changing its output when the spikes occur and makes it converge against  $\bar{P}_B$ , this demonstrate that the trajectory plan-



ner is working as desired and that the feedback of the obtained position to the trajectory planner makes the close loop correct the errors. The spikes are made by the inverse manipulator kinematics, which are calculating the joint angles, these are illustrated on Figure 18.6 for the LF leg.

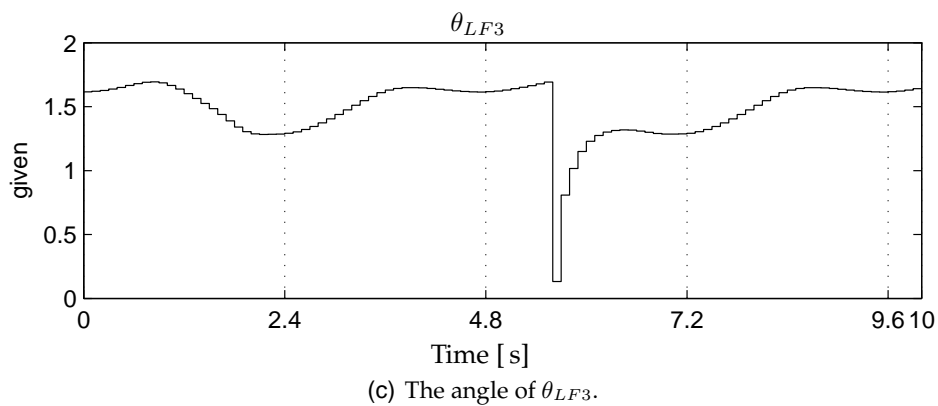
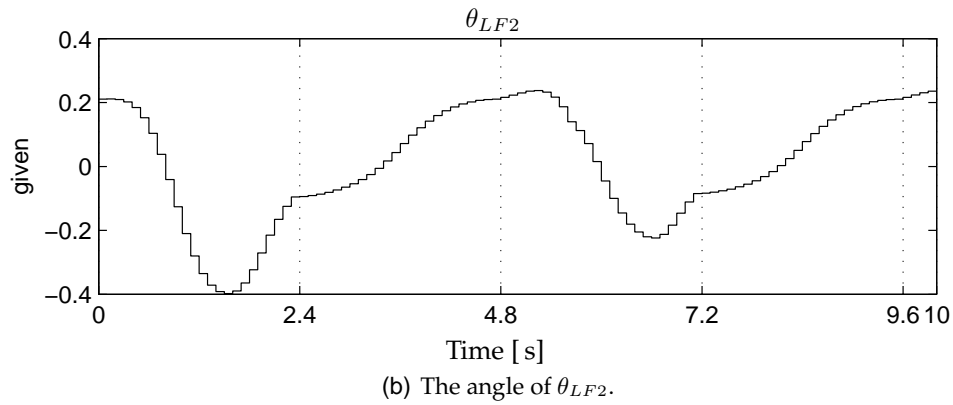
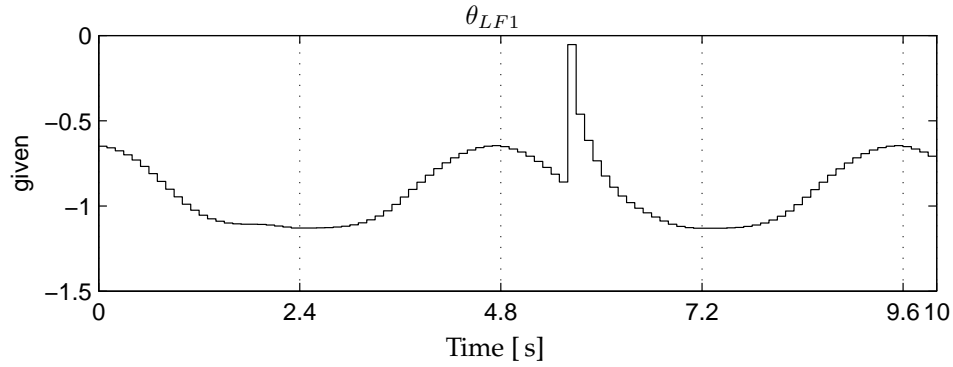


FIGURE 18.6: The obtained joint angles in the walk test.

There is a spike in  $\theta_{LF1}$  and  $\theta_{LF3}$  at 5.7[s], as there were in  $\bar{P}_B$ . The joint angles are calculated in the following fashion;  $\theta_{LF2}$ ,  $\theta_{LF3}$  and  $\theta_{LF1}$  as the inverse manipulator

kinematics calculations were derived in Appendix B. Therefore the spike in  $\theta_{LF3}$  will be investigated first because it is the first calculated angle with an error.  $\theta_{LF3}$  has periodic sequence where the angle at 5.7[s] had been calculated without a spike earlier in the sequence, but this time the steering controller is changing the desired position as illustrated on Figure 18.4, this moves the desired position outside the working range, the calculated joint angle is therefore corrected to an angle inside the working range, which in this case is wrong. This can be prevented by using the  $(i - 1)$  angle to validate how the calculated angle should be corrected to get inside the working range securing that the position is achievable before calculating the angle.  $\theta_{LF3}$  is used in the calculation of  $\theta_{LF1}$  in Equation (10.1), therefore  $\theta_{LF1}$  will become wrong when  $\theta_{LF3}$  is wrong.

The angle of the body  $\vartheta_{CM}$  in the walk test is illustrated on Figure 18.7.

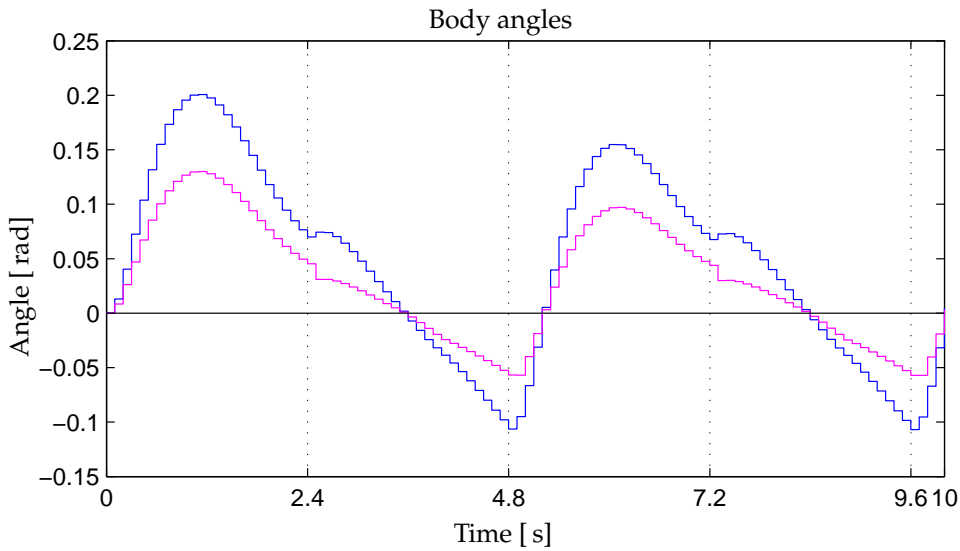


FIGURE 18.7: An illustration of the how the body angle is changing during walk. The blue line represents rotation around the x-axis, the magenta line rotation around the y-axis and the black line rotation around the z-axis of coordinate system  $\{1\}$ .

The body angle is calculated around the axis of coordinate system  $\{1\}$ , since turning is disregarded in Chapter 2 the rotation around the z-axis is zero. The rotation around the x and y axis does not have similar periods with constant offset, therefore the walk is tested again over a longer period. This test demonstrated that the rotations of the body were settled from 4.8[s], therefore these test results will be redundant to the earlier demonstrated and therefore not be illustrated. The NBT should fall forward in the trot locomotion as described in Chapter 16, this concurs with that the rotation around the y-axis has an offset of 0.04[rad]. The rotation around the x-axis also has an offset of 0.04[rad], which gives

an slight rotation to the right side of the NBT. This can be connected to the curve shape of the rotations around the x and y axis, which changes direction at every period shift. The period shifts from  $F_D$  2 to 3 in 2.4[s] and 7.2[s] changes in one fashion and 3 to 2 in 4.8[s] and 9.6[s] changes in another fashion. This can be because of the movement of the legs and the location of the CM or an error in the implementation of the sensor. The sensor was not the scope of this project and therefore was kept simple. The rotations of the body is calculated from the  $(i - 1)$  value of  $\bar{\theta}_{N,m}$  and  $\bar{\theta}_{N,m}$  in Equation (13.11) and the location of the CM. The location of the CM is illustrated on Figure 18.8.

The changes in the z component are similar in each period, except in the period were the joint angles get outside the working range, this shows that the legs are lifted similarly for both LP. The x component has a sinus shape with an offset in positive direction, which demonstrate that the fall is in forward direction as intended in the design of the walk in the path planner and steering control in Chapter 15 and Chapter 16. The error in the joint angles for the LF leg are also influencing on CM's location, which demonstrates that the leg movement influences on where the CM is located, as they shall. The y component is the sideways location of the CM, this is not moving considerably, note the scale on the figure. It is settling approximately without an offset. The wave top ending at 7.2[s] is a result of the steering controller correcting the movement as illustrated on Figure 18.4.

The body angle is inside the boundaries given by  $\theta_D$ , which shows that the rotations of the body is a limited circle as intended in the design of the steering controller. The rotation of the body also demonstrates that the controller is capable of stopping the falls of the inverted pendulum construction.

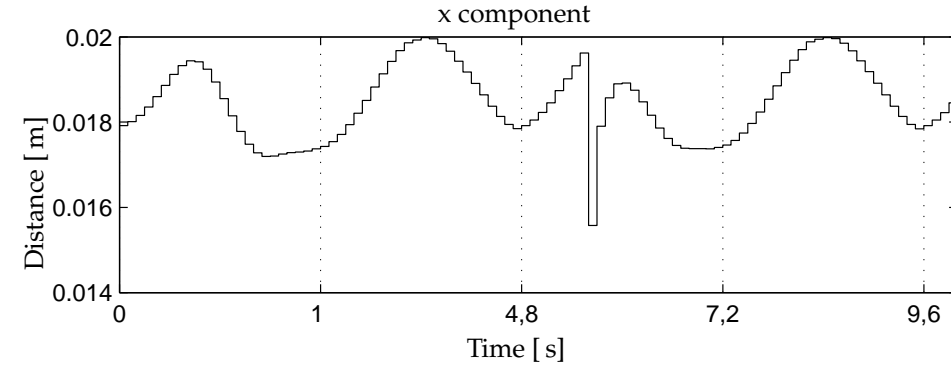
(Note: The visualization of the test are recorded as a video clip, this can be found in /simulation/video/system\_test2.avi.)

## 18.3 STABILITY WITH WHITE NOISE

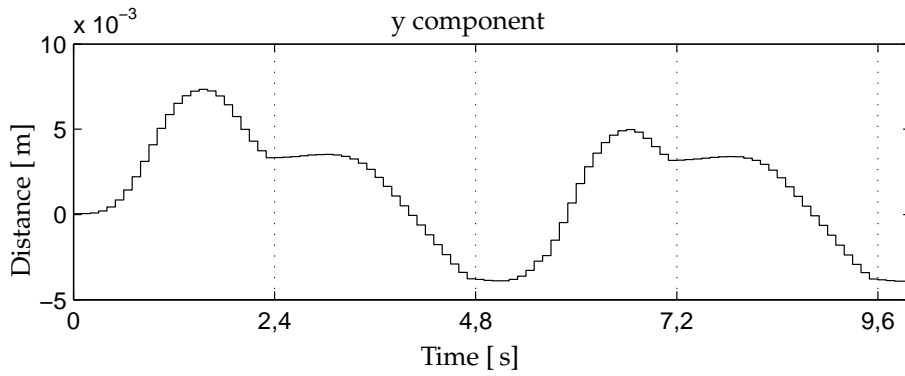
The stability of the combined system, with white noise added to the output joint angles, which will be tested as in section 18.2. The white noise signal added to the joint angles is illustrated on Figure 18.9.

The phase between the legs controlled by  $F_D$  and the position vectors there, is compared to find out if the desired positions are obtained,  $\bar{P}_B$  and  $\bar{P}_{N,m}$  are illustrated on Figure 18.10.

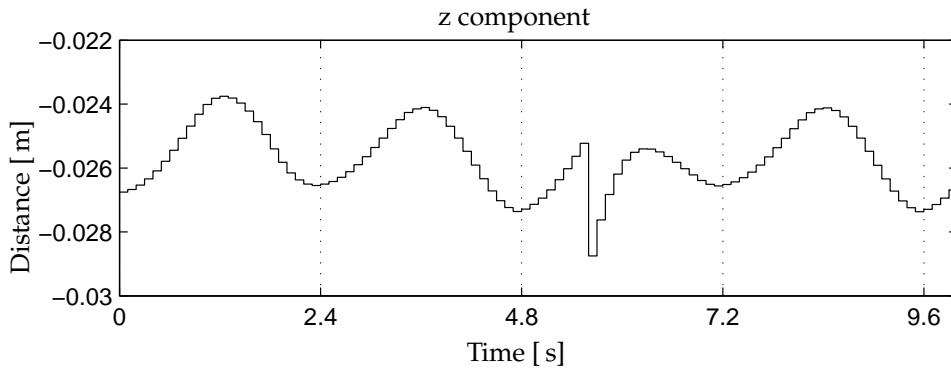
$F_D$  shift value at 2.4[s], 6.8[s] and 7.2[s], the first shift is at the correct time, the second uses longer time and the third shift to fast.  $\bar{P}_B$  shifts to the next position when  $F_D$



(a) The x component of the position vector.



(b) The y component of the position vector.



(c) The z component of the position vector.

**FIGURE 18.8:** The location of the CM when testing the walk. The position vector is from coordinate system  $\{1\}$  to the CM.

changes and corrects the movements to catch the falls, as it is supposed to.  $\bar{P}_m$  is calculated from  $\theta_{N,m}$  which has an overlaying white noise, which change the  $\bar{P}_m$  from the signal without noise illustrated on Figure 18.3. Even though the noise is  $\bar{P}_m$  over time converging toward  $\bar{P}_B$  wherefore  $F_D$  shift value and the LP in swing and stand phase are shifted. The converging is because the *Trajectory planner* is correcting the trajectory

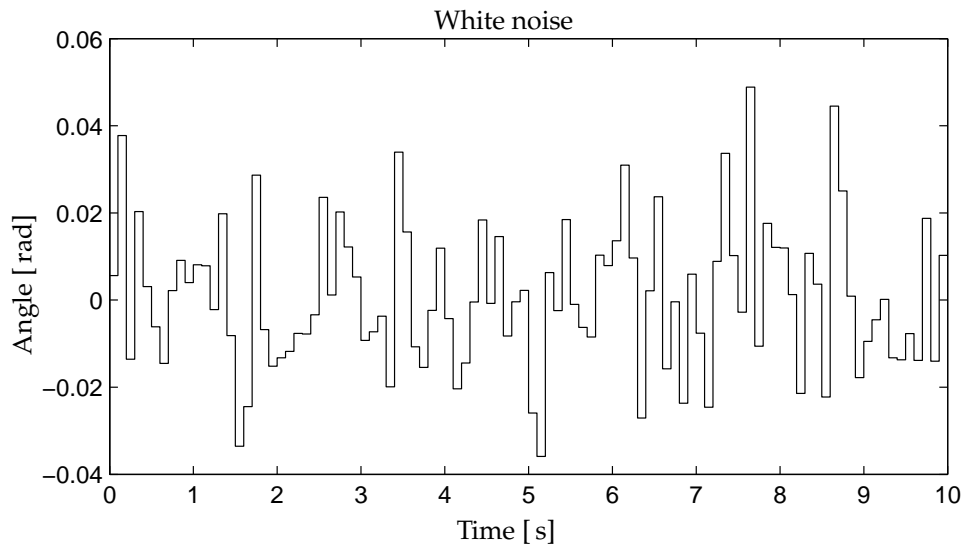


FIGURE 18.9: An illustration of the white noise added to the body angles.

for each of the feet dependent on the  $(i - 1)$  value of  $\bar{P}_m$ ,  $\bar{P}_B$  and the time to obtain the position.

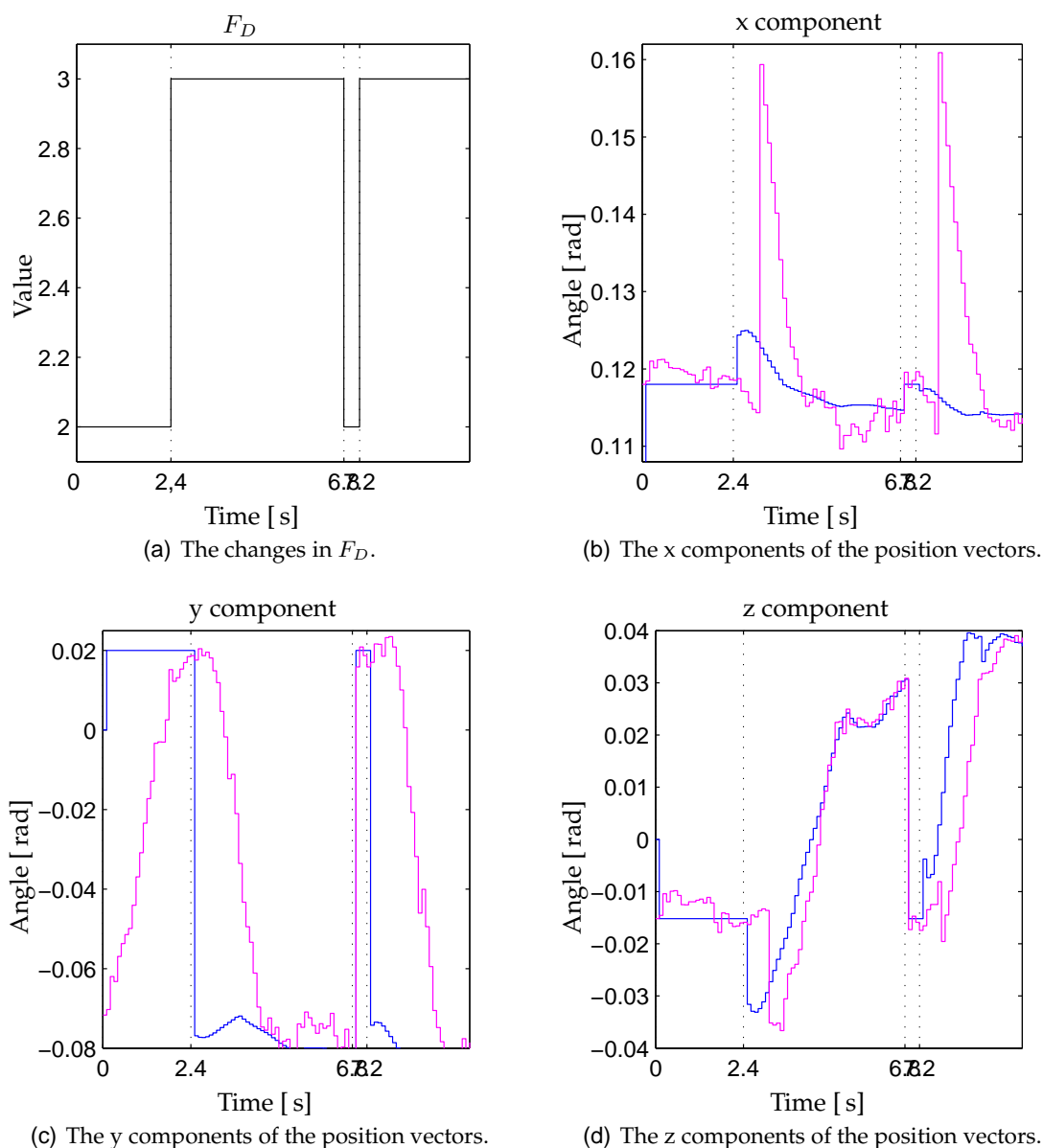
The fast shift of  $F_D$  is because the timer resets when the time for the period is used, and the state machine in the *supervisor* change to either state C or E as the conditions for shift 3 and 5 are, as described in Chapter 5. These state check if the feet are in the desired position and shifts  $F_D$ .

The joint angles for the RF leg are illustrated on Figure 18.11. The blue line is without noise and the magenta line with noise.

The joint angles have deviations from the joint angles without noise; this demonstrates that the noise has an influence on the angles that are fed back to the system. A small angle deviation gives a larger deviation in the position vector, because of the trigonometrical relations for a triangle where the angle is changed. The joint angles are used to calculate the rotations of the body, this is illustrated on Figure 18.12.

The calculated body angles are also deviating from the ones without noise that are illustrated on Figure 18.7. The body angle is not settled and will not do it, when  $F_D$  not is changing periodic. The body angle stops decreasing from 5.1 to 6.8[s] and 8.3 to 10.0[s] which are errors.

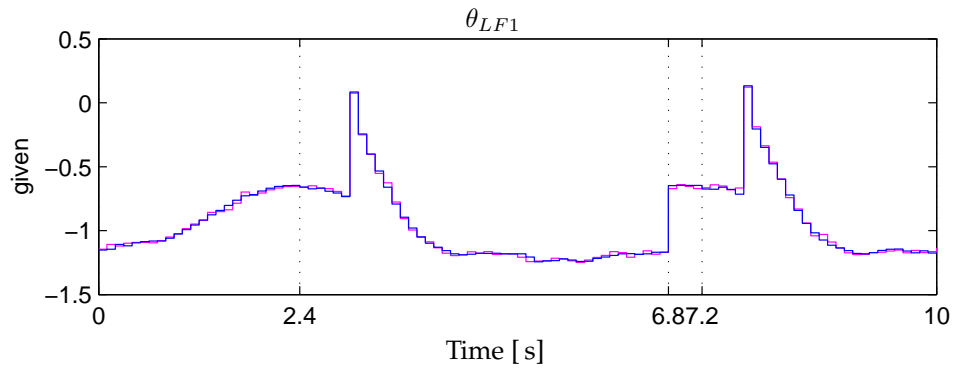
The noise has an amplitude of 0.08[rad] and the changes in joint angles have amplitudes of approximately 0.92[rad], 0.50[rad] and 0.42[rad], for  $\theta_{LF1}$ ,  $\theta_{LF2}$  and  $\theta_{LF3}$ , respectively, before the noise is added, this is measured on Figure 18.6. The signal to noise ratio are 11.5, 6.25 and 5.25, respectively, these ratios shows that the system is robust



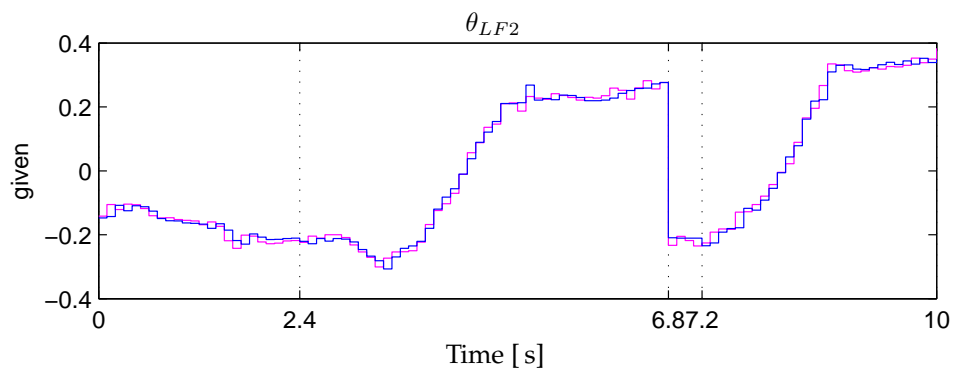
**FIGURE 18.10:** Test results of walking with noise on the feedback. The blue line represents the component of  $\bar{P}_B$ , the magenta line the component of  $\bar{P}_m$  and the black line  $F_D$ .

against disturbances.

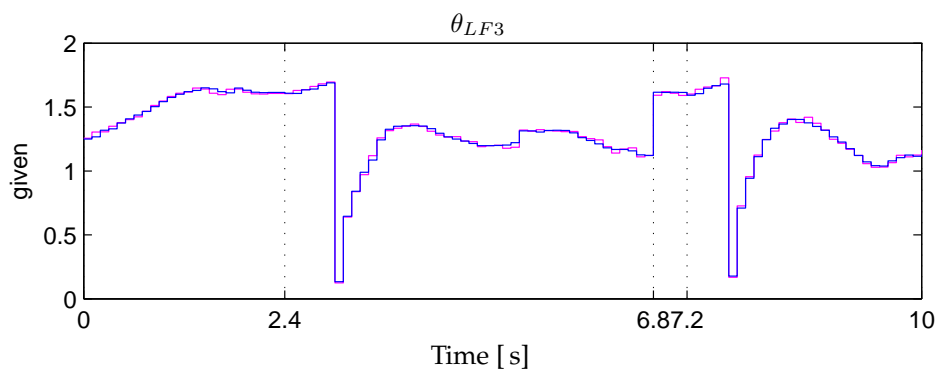
(Note: The visualization of the test are recorded as a video clip, this can be found in `/simulation/video/system_test3.avi`.)



(a) The angle of  $\theta_{LF1}$ .

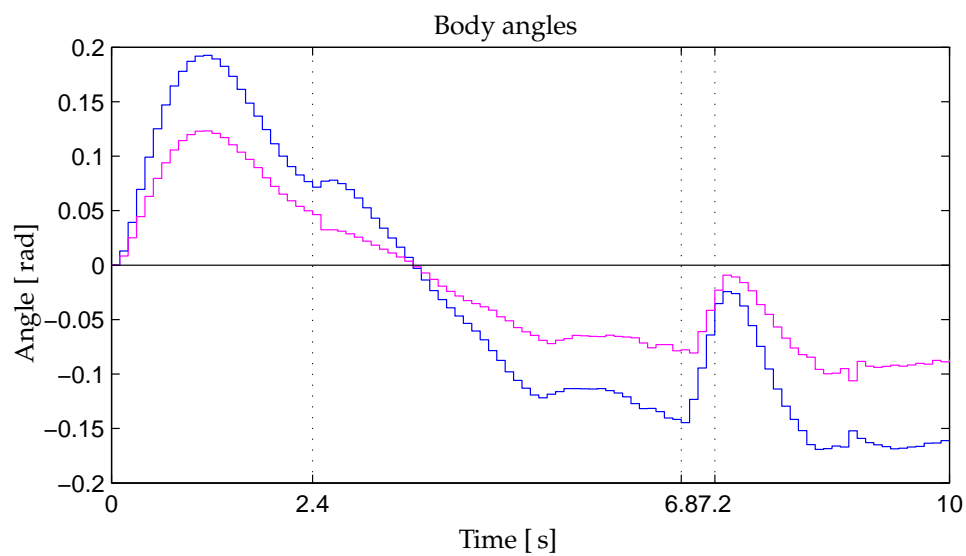


(b) The angle of  $\theta_{LF2}$ .



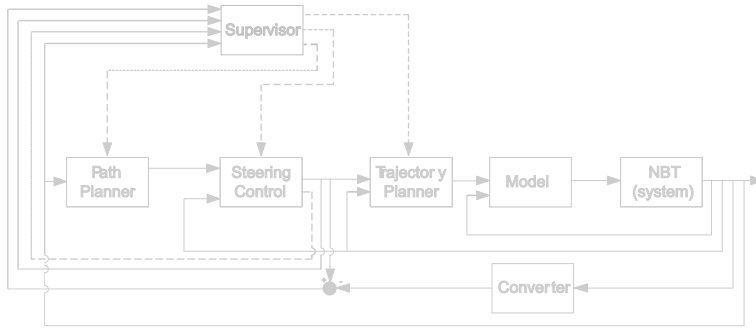
(c) The angle of  $\theta_{LF3}$ .

FIGURE 18.11: The obtained joint angles in the walk test with and without noise. The blue line is without noise and the magenta line is with noise.



**FIGURE 18.12:** An illustration of the how the body angle is changing during walk with feedback signals with white noise added. The blue line represents rotation around the  $x$ -axis, the magenta line around the  $y$ -axis and the black line around the  $z$ -axis of coordinate system  $\{1\}$ .





## CONCLUSION

This project concerns the control of an autonomous robot dog, the Newcastle Bull Terrier (NBT). The primary goal of the project was to make the NBT walk, while avoiding falls. This was limited to walk along a straight line with a velocity of at least  $0.1467 \text{ [m/s]}$  because others had achieved this velocity. To achieve this, the *quadruped walking*, *NBT* and *Central Pattern Generator (CPG)* were analyzed, which made the basis for construction of a control system, consisting of a; *Supervisor*, *Path planner*, *Steering controller*, *Trajectory planner* and *Model*.

The analysis of the NBT found the sizes, weights, sensors etc. for the robot, this made the basis for deriving of the model. The model consisted of three parts; *manipulator kinematics model*, *inverse manipulator kinematics model* and *manipulator dynamical model*. The model was structurally verified both with constant and varying inputs, to prove the general structure of the model.

The phase between the legs and the overall choices of when to start walking, stopping, desired velocity, etc. were controlled by the *supervisor* which were implemented as a discrete event state machine.

The walk were chosen to trot with the feet moving along a parabolic curve when they are moved forward, the balance of the NBT were found to be dynamics, therefore the robot always were falling as an inverted pendulum. The steering controller were designed to use the falls to generate the forward velocity by stopping them when the feet are put on the floor, thereby controlling the rotations of the NBT's body inside a limited circle.

Animals use their CPG to make the paths and trajectories for their movements, these were designed and implemented as two independently blocks. The *path planner* controlled where the feet should be moved in the current phase of the walk as trot gait defines, together with the step length and time to move the legs. The step length is designed to hold the center of pressure close to the line between the supporting legs to minimize the acceleration of the fall of the inverted pendulum construction. The *trajectory planner* controlled where the feet should be at every iteration, to obtain the position from the *path*

---

*planner* at the end of the period.

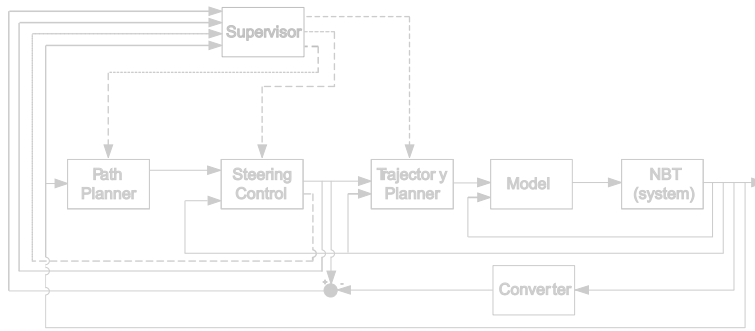
To test the control system, some sensors had to be designed and implemented; these were kept as simple as possible because they were not the scope of the project. A sensor to calculate the angular acceleration, velocity and position were implemented. A converter from joint angles to position vector was also implemented by use of the derived *manipulator kinematics model*

The control system was tested by simulating walk both with and without noise on the feedback signals. Without noise the body angle did change with a repeating pattern over a period of a step for both Leg Pair (LP), it had a minor angle offset around the longitudinal and latitudinal axis in the body. The periods were shifted at the correct time and the steering controller corrected the position of the feet so the falls were stopped. The inverse manipulator kinematics corrected a joint angle wrongly once in the simulation, this can be avoided by securing that the desired position is achievable. The trajectory planner makes the achieved position of the feet converge against the desired position from the steering controller, it is also makes the error converge against the desired position.

With noise the body angle did also change with a repeating pattern as without the noise, but the offset increased. This can not be concluded on because the LP in the different phases of trot were not shifted correct, this means that the inverted pendulum construction of the body keeps falling. The phases were not shifted correctly because the error between the desired position and the feedback of the achieved position was larger than the accepted  $0.001[\text{m}]$ , when the error became smaller than the accepted maximum, the period was shifted. The position feedback was calculated from the obtained joint angles with an overlaying white noise, which is the reason that the error became larger than accepted. The simulation of the walk was performed with a period time of  $0.6[\text{s}]$  for every step, this gives an achieved velocity of  $1.5283[\text{m/s}]$ .

A 3D-game simulator was modified with a new field and sky, for visualization of the simulation results. The prototype was constructed as three dimensional objects, that were connected to form the NBT, and the simulated walk with and without noise were illustrated in 3D environment. Videos of the simulations can be found on the CD.

The goal of the project was to make the NBT walk without falling, this was achieved without noise on the feedback signals, with noise on the feedback signals it was achieved inside the simulation period of  $2.5[\text{s}]$ , but over time the NBT will fall because the legs were not shifted correctly. There was also a claim to achieving a velocity of at least  $0.1467[\text{m/s}]$ , the walk was simulated with a velocity of  $1.5283[\text{m/s}]$ , wherefore that claim was achieved.



20

## PERSPECTIVES

It would be desirable to validate if the demanded positions were achievable before the joint angles were calculated, to secure that the joint angles were correct, and thereby remove the twitches with the legs when the position is unachievable.

The load from the sensors in the motors, will give a spike when the feet are put on the floor, this can be used to minimize the influence of the noise on the feedback signals by increasing the comparison of the position error and time, with the load and weight the three sensors signal.

When these two things, mentioned above, are made the system can be implemented on the real NBT, and an accept test can be preformed. It will however be advantages to design walk controllers to control the movement of each joint in closed loop, to insure that the legs move as desired.



# BIBLIOGRAPHY

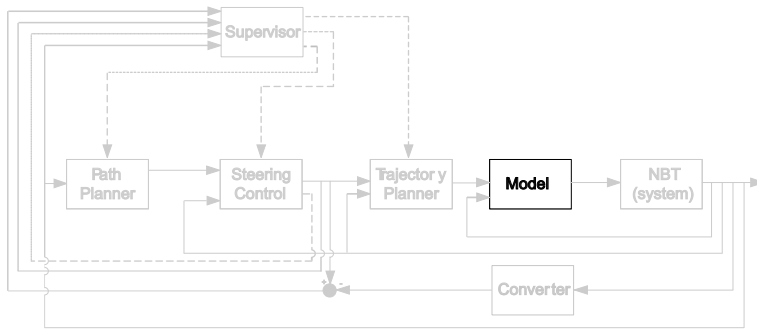
- [AMA03] C. Chevallereau A. Muraro and Y. Aoustin.  
Optimal trajectories for a quadroped robot with trot, amble, curvet gaits for two energetic criteria.  
*Multibody System Dynamics* 9, 2003.
- [BG96] M. Bennani and F. Giri.  
Dynamic modelling of a four-legged robot.  
*Journal of Intelligent and Robotic Systems* 17, 1996.
- [BHS00] S. Pham B. Hengst, D. Ibbotson and C. Sammut.  
*The UNSW United 2000, Sony legged Robot Software System.*  
Technical report, School of Computer Science and Engineering, University of New South Wales, Sydney NSW 2052, Australia, November 2000.  
URL <http://www.cse.unsw.edu.au/~robocup/2002site/2000PDF.zip>.  
Downloaded 20-02-2007.
- [Com] RoboCup Technical Committee.  
*RoboCup Four-Legged League Rule Book.*  
Technical report.  
URL <http://www.tzi.de/4legged/pub/Website/Download/Rules2006.pdf>.  
Downloaded 23-07-2006.
- [Cra05] John J. Craig.  
*Introduction to Robotics Mechanics and Control, 3th. ed.*  
Prentice Hall, 2005.  
ISBN 0-13-123629-6.
- [Eur] Sony Entertainment Robot Europe.  
*Important announcement.*  
Technical report.  
URL <http://support.sony-europe.com/aibo/index.asp>.  
Viewed 23-07-2006.
- [GFFP02] Abbas Emami-Naeini Gene F. Franklin and J. David Powell.

- Feedback Control of Dynamic Systems, 4th. ed.*  
Prentice Hall, 2002.  
ISBN 0-13-032393-4.
- [HB00] Geoff Hawker and Martin Buehler.  
Quadruped trotting with passive knees - design, control, and experiments.  
*IEEE*, 2000.
- [JFK95] Masamichi Sakaguchi Junji Furusho, Akihito Sano and Eichi Koizumi.  
Realization of bounce gait in a quadruped robot with articular-joint-type legs.  
*IEEE*, 1995.
- [LB02] M. Anthony Lewis and George A. Bekey.  
Gait adaption in a quadruped robot.  
*Autonomous Robots 12*, 2002.
- [MZF04] T. Iwasaki M. Zheng and W. O. Friesen.  
System approach to modeling the neuronal cpg for leech swimming.  
*IEEE EMBS*, 2004.
- [Oga70] Katsunhiko Ogata.  
*Modern Control Engineering, 2th. ed.*  
Prentice Hall, 1970.  
ISBN 0-13-589128-0.
- [PG] 2006 Project Group 733.  
*The 3D Visualiser Manual*.  
Technical report.  
URL <http://www.control.aau.dk/~06gr733/3D/manual.pdf>.  
Viewed 05-02-2007.
- [SB00] Raymond A. Serway and Robert J. Beichner.  
*Physics - For Scientists and Engineers with modern physics*.  
Harcourt College Publishers, 2000.  
ISBN 0-03-022657-0.
- [Tri] Tribotix.  
*User manual, Dynamixel DX-113, DX-116, DX-117*.  
Technical report.  
URL [http://www.tribotix.info/Downloads/Dynamixels/DX%20series\(english\).pdf](http://www.tribotix.info/Downloads/Dynamixels/DX%20series(english).pdf).  
Downloaded 13-09-2006.

- [TwY99] Tae-wan and Yong-San Yoon.  
Dynamic instant gait stability measure for quadruped walking robot.  
*Robotica* 17, 1999.
- [UoI] Turkey University of Istanbul.  
Control system architecture for an autonomous quadruped robot.







A

## MANIPULATOR KINEMATICS

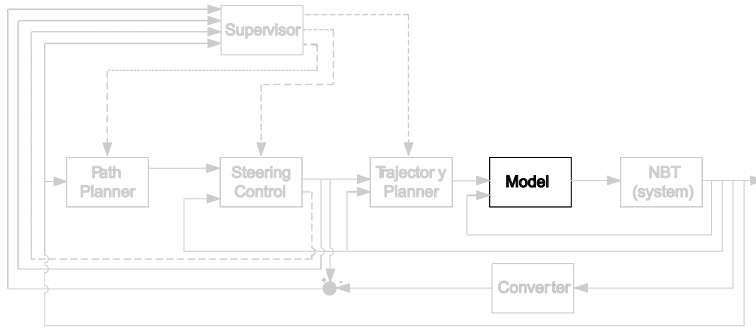
*This appendix concerns the modeling of the manipulator kinematics and will contain angles and lengths between the different coordinate systems. These are described in Chapter 9.*

Table A.1 on the next page contains the angles and lengths between the coordinate systems. The extra coordinate systems there are introduced to make the rotations possibly, will also be mentioned in the table.

---

i	$\alpha_i$ [rad]	$L_i$ [m]	$d_i$ [m]	$\theta_i$ [rad]
Ha	0	0	0	$-\pi/2$
Hb	$\pi/2$	0	$0.1700/2 + 0.0360/2$	$\theta_H$
LF1	$-\pi/2$	$0.1700/2$	$0.1088/2$	$\pi/2 + \theta_{LF1}$
LF2	$-\pi/2$	0.0110	0	$\theta_{LF2}$
LF3	$\pi/2$	0.0656	0	$\theta_{LF3}$
LF4a	0	0.0866	0	$-\pi/2$
LF4b	$\pi/2$	0.0110	0	0
RF1	$-\pi/2$	$0.1700/2$	$-0.1088/2$	$\pi/2 + \theta_{RF1}$
RF2	$-\pi/2$	0.0110	0	$\theta_{RF2}$
RF3	$\pi/2$	0.0656	0	$\theta_{RF3}$
RF4a	0	0.0866	0	$-\pi/2$
RF4b	$\pi/2$	0.0110	0	0
LR1a	0	$-0.0740/2$	0	$-\pi/2$
LR1b	$\pi/2$	0	0	$\theta_{LR1}$
LR2	$-\pi/2$	0.0110	0	$\theta_{LR2}$
LR3	$\pi/2$	0.0656	0	$\theta_{LR3}$
LR4a	0	0.0866	0	$-\pi/2$
LR4b	$\pi/2$	$-0.0110$	0	0
RR1a	0	$0.0740/2$	0	$-\pi/2$
RR1b	$\pi/2$	0	0	$\theta_{RR1}$
RR2	$-\pi/2$	0.0110	0	$\theta_{RR2}$
RR3	$\pi/2$	0.0656	0	$\theta_{RR3}$
RR4a	0	0.0866	0	$-\pi/2$
RR4b	$\pi/2$	$-0.0110$	0	0

TABLE A.1: Link parameters for the coordinate systems placed in the NBT.



B

## INVERSE MANIPULATOR KINEMATICS

*This appendix contains the deriving of the formulas to calculate the joint angles in the legs, for Chapter 9.*

An automatic method shall be derived to calculate all the 12 joint angles in the legs. This will be done by finding geometrical relations where from the angles can be derived. The method shall only be able to calculate the angles inside the working range of the joints described in Chapter 10. Aside from the direction of the displacement in the knees is the legs similar, therefore will the method be derived for a neutral leg and differences between the legs will be described. The neutral leg that the method will be derived for, is chosen to be the LF leg.

The UL, LL,  $\theta_{LF3}$  and foot is moving in a plan that is rotated out from the plan that  $\theta_{LF1}$  is working in. The rotation is around the axis of  $\theta_{LF2}$ . When  $\theta_{LF2}$  is zero is the plans coinciding. The plan that  $\theta_{LF1}$  rotates in is donated *plan A* and the plan rotated by  $\theta_{LF2}$  is donated *plan B*, this is illustrated on Figure B.1.

The coordinate system placements can be found in Chapter 9 and 10.

$\theta_{LF2}$  is the only one of the three joint angles that never has the same rotation direction as the other two joint angles, this will be used. As illustrated on Figure B.2(a) are the size of  ${}^{LF0}_{LF4}P_{d,z}$  only dependent on the length in plan B, therefore is it only depends on  $\theta_{LF2}$ ,  $\theta_{LF3}$  and constants. The length in plan B is illustrated in Figure B.2(b).

$\theta_{LF3A}$  is the angle directly from the knee to the foot this angle is equal to  $\theta_{LF3}$  plus the offset angle given by the displacement in the knee, this is illustrated on Figure B.3.

$P_4$  is the position of the foot and  $P_5$  is the position of  $P_4$  projected in to the direction of  $L_{LF2 \rightarrow LF3}$ , which then can be projected into the direction of  ${}^{LF0}_{LF4}P_{d,z}$ . The lengths  $L_{LF1 \rightarrow LF2}$ ,  $L_{LF2 \rightarrow LF3}$  and  $L_{LF3 \rightarrow LF4}$  will in the rest of this appendix be referred to as  $l_1$ ,  $l_2$  and  $l_3$ , respectively, to make the calculations easier to foresee. With the same intention will  ${}^{LF0}_{LF4}P_{d,x}$ ,  ${}^{LF0}_{LF4}P_{d,y}$  and  ${}^{LF0}_{LF4}P_{d,z}$  be referred to as  $P_x$ ,  $P_y$  and  $P_z$ , respectively. The

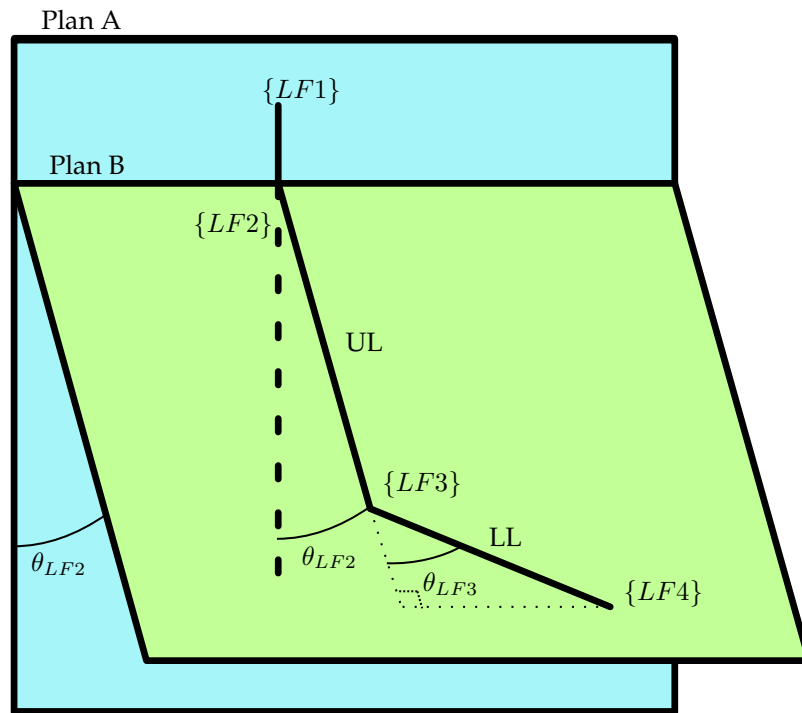


FIGURE B.1: Illustration of how plan B is rotated in collation with plan A.

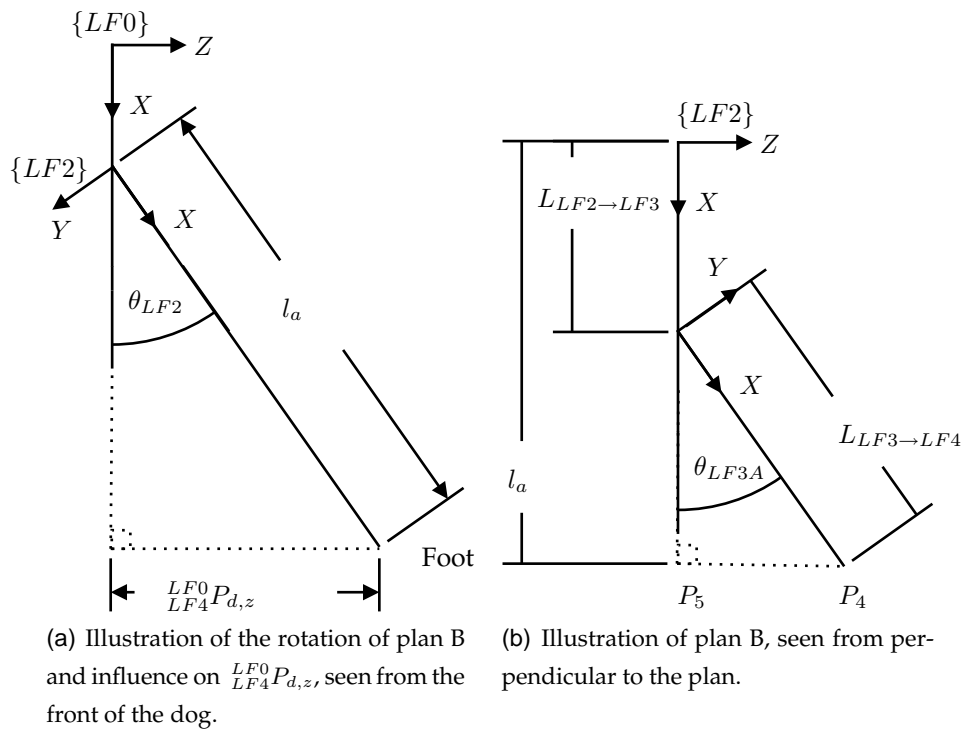


FIGURE B.2: Illustration of the rotation of plan B and length in the plan.

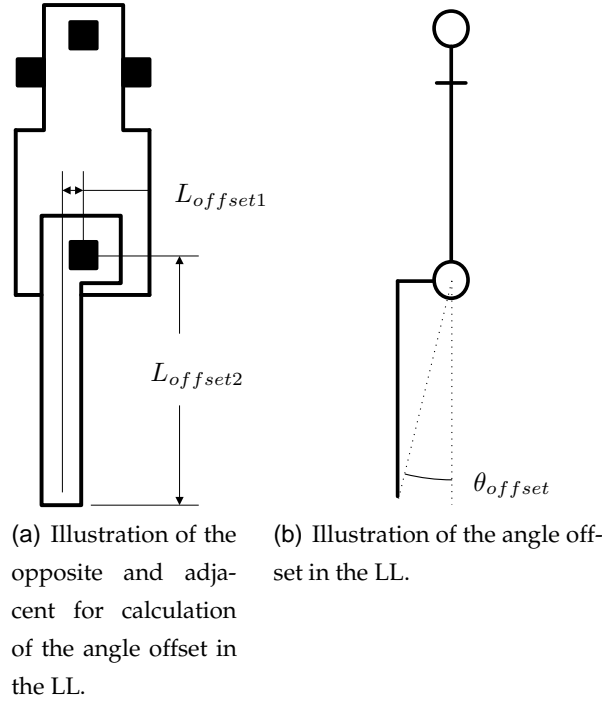


FIGURE B.3: The illustration is viewed from the left side of the NBT.

projection of  $l_3$  into the direction of  $l_2$  is calculated in Equation (B.1).

$$\begin{aligned} P_z &= l_a \cdot \sin(\theta_{LF2}) \\ &= (l_2 + l_3 \cdot \cos(\theta_{LF3A})) \cdot \sin(\theta_{LF2}) \end{aligned} \quad (B.1)$$

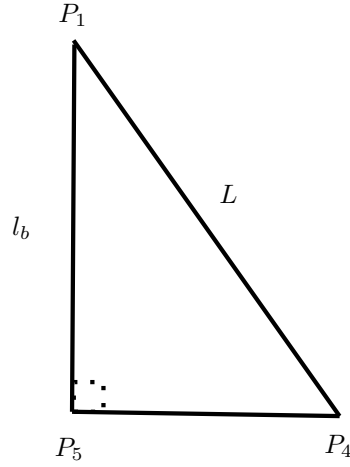
Now the length from origo of  $\{LF0\}$  to  $P_5$  can be found, this length will be donated  $l_b$  and is found in Equation (B.2) by use of the cosines relation.

$$l_b^2 = l_1^2 + (l_2 + l_3 \cdot \cos(\theta_{LF3A}))^2 + 2 \cdot l_1(l_2 + l_3 \cdot \cos(\theta_{LF3A})) \cos(\theta_{LF2}) \quad (B.2)$$

There now will be made a RAT from origo of  $\{LF0\}$  to  $P_4$  and  $P_5$ , this is illustrated on Figure B.4. Origo of  $\{LF0\}$  is donated  $P_1$  and  $L$  is the length from  $P_1$  to the foot.

The length of  $L$  is found by use of Pythagoras theorem in Equation (B.3a).

$$\begin{aligned} L^2 &= P_x^2 + P_y^2 + P_z^2 \\ &= l_b^2 + l_3^2 \cdot \sin^2(\theta_{LF3A}) \\ &= l_1^2 + (l_2 + l_3 \cdot \cos(\theta_{LF3A}))^2 + 2 \cdot l_1(l_2 + l_3 \cdot \cos(\theta_{LF3A})) \cos(\theta_{LF2}) \\ &\quad + l_3^2 \cdot \sin^2(\theta_{LF3A}) \\ &= l_1^2 + l_2^2 + l_3^2 + 2l_2l_3 \cos(\theta_{LF3A}) + 2l_1(l_2 + l_3 \cos(\theta_{LF3A})) \cos(\theta_{LF2}) \\ &= l_1^2 - l_2^2 + l_3^2 + 2l_2(l_2 + l_3 \cos(\theta_{LF3A})) + 2l_1(l_2 + l_3 \cos(\theta_{LF3A})) \cos(\theta_{LF2}) \end{aligned} \quad (B.3a)$$



**FIGURE B.4:** Illustration of the RAT made between  $P_1$ ,  $P_4$  and  $P_5$ .

Equation (B.1) is changed in Equation (B.4) and inserted into Equation (B.3a), this is done in Equation (B.5).

$$(l_2 + l_3 \cdot \cos(\theta_{LF3A})) = \frac{P_z}{\sin(\theta_{LF2})} \quad (\text{B.4})$$

$$L^2 = l_1^2 - l_2^2 + l_3^2 + \frac{2l_2P_z}{\sin(\theta_{LF2})} + \frac{2l_1P_z}{\sin(\theta_{LF2})} \cos(\theta_{LF2}) \quad (\text{B.5a})$$

$$0 = (l_1^2 - l_2^2 + l_3^2 - L^2) \sin(\theta_{LF2}) + 2l_2P_z + 2l_1P_z \cos(\theta_{LF2}) \quad (\text{B.5b})$$

$$-2l_2P_z = (l_1^2 - l_2^2 + l_3^2 - L^2) \sin(\theta_{LF2}) + 2l_1P_z \cos(\theta_{LF2}) \quad (\text{B.5c})$$

The sinus addition formula in Equation (B.6) will be used to simplify the equation by defining tangents to be as in Equation (B.7)

$$\sin(\alpha + \beta) = \sin(\alpha) \cos(\beta) + \cos(\alpha) \sin(\beta) \quad (\text{B.6})$$

$$\tan(\varrho) = \frac{2l_1P_z}{l_1^2 - l_2^2 + l_3^2 - L^2} \quad (\text{B.7})$$

This will be used on Equation (B.5).

$$\sin(\varrho + \theta_{LF2}) = \frac{(l_1^2 - l_2^2 + l_3^2 - L^2)}{\sqrt{((l_1^2 - l_2^2 + l_3^2 - L^2)^2 + 4l_2^2 P_z^2)}} \sin(\theta_{LF2}) \quad (\text{B.8a})$$

$$+ \frac{2l_1 P_z}{\sqrt{((l_1^2 - l_2^2 + l_3^2 - L^2)^2 + 4l_2^2 P_z^2)}} \cos(\theta_{LF2})$$

$$= \frac{-2l_2 P_z}{\sqrt{((l_1^2 - l_2^2 + l_3^2 - L^2)^2 + 4l_2^2 P_z^2)}} \quad (\text{B.8b})$$

$$\theta_{LF2} = -\arcsin\left(\frac{2l_2 P_z}{\sqrt{((l_1^2 - l_2^2 + l_3^2 - L^2)^2 + 4l_2^2 P_z^2)}}\right) \quad (\text{B.8c})$$

$$- \arctan\left(\frac{2l_1 P_z}{l_1^2 - l_2^2 + l_3^2 - L^2}\right)$$

When  $\theta_{LF2}$  is known can the method to calculate  $\theta_{LF3A}$  be derived from Equation (B.4).

$$l_3 \cdot \cos(\theta_{LF3A}) = \frac{P_z}{\sin(\theta_{LF2})} - l_2 \quad (\text{B.9a})$$

$$\cos(\theta_{LF3A}) = \frac{\frac{P_z}{\sin(\theta_{LF2})} - l_2}{l_3} \quad (\text{B.9b})$$

$$\theta_{LF3A} = \arccos\left(\frac{\frac{P_z}{\sin(\theta_{LF2})} - l_2}{l_3}\right) \quad (\text{B.9c})$$

The calculation of  $\theta_{LF3A}$  in Equation (B.9c) is not valid for  $\theta_{LF2} = 0$  because there will be divided by zero, therefore is a method to calculate  $\theta_{LF3A}$  when  $\theta_{LF2} = 0$  derived in Equation (B.10).

$$L^2 = (l_1 + l_2)^2 + l_3^2 - 2(l_1 + l_2)l_3 \cos(\theta_{LF3A}) \quad (\text{B.10a})$$

$$2(l_1 + l_2)l_3 \cos(\theta_{LF3A}) = (l_1 + l_2)^2 + l_3^2 - L^2 \quad (\text{B.10b})$$

$$\cos(\theta_{LF3A}) = \frac{(l_1 + l_2)^2 + l_3^2 - L^2}{2(l_1 + l_2)l_3} \quad (\text{B.10c})$$

$$\theta_{LF3A} = \arccos\left(\frac{(l_1 + l_2)^2 + l_3^2 - L^2}{2(l_1 + l_2)l_3}\right) \quad (\text{B.10d})$$

As earlier mentioner is  $\theta_{LF3}$  equal to  $\theta_{LF3A}$  plus an offset angle, as illustrated in Figure B.3.  $\theta_{LF3,offset}$  is calculated in Equation (B.11a) and  $\theta_{LF3}$  is calculated in Equation (B.11b).

$$\theta_{LF3,offset} = \arctan\left(\frac{L_{offset1}}{L_{offset2}}\right) \quad (\text{B.11a})$$

$$\theta_{LF3} = \theta_{LF3A} + \theta_{LF3,offset} \quad (\text{B.11b})$$

Finally a method for calculating  $\theta_{LF1}$  will be derived, by use of that  $\theta_{LF2}$  and  $\theta_{LF3}$  are known. The location of the foot with  $\theta_{LF1}$  in its initial angle is projected into plan A, the result of this projection is illustrated together with the desired position of the foot on Figure B.5. There are inserted an isosceles triangle between the two points and the center of the coordinate system.

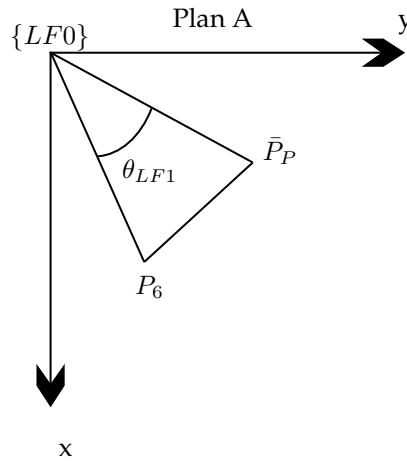


FIGURE B.5: Illustration of how a triangle can be made to calculate  $\theta_{LF1}$ .

The method to calculate the angle of  $\theta_{LF1}$  is derived in Equation (B.12). The Equation finds the angle from the x axis of the coordinate system to both of the lines from the center of the coordinate system to  $P_6$  location and  $\bar{P}_P$ . Angles then are subtracted from each other.

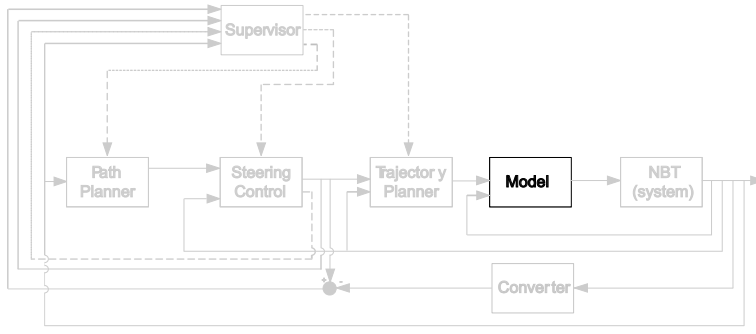
$$P_{6,x} = (l_2 + l_3 \cos(\theta_{LF3A})) \cos(\theta_{LF2}) + l_1 \quad (\text{B.12a})$$

$$P_{6,y} = l_3 \sin(\theta_{LF3A}) \quad (\text{B.12b})$$

$$\theta_{LF1} = \arctan\left(\frac{P_{6,y}}{P_{6,x}}\right) - \arctan\left(\frac{P_{P,y}}{P_{P,x}}\right) \quad (\text{B.12c})$$

$$= \arctan\left(\frac{l_3 \sin(\theta_{LF3A})}{(l_2 + l_3 \cos(\theta_{LF3A})) \cos(\theta_{LF2}) + l_1}\right) - \arctan\left(\frac{P_{P,y}}{P_{P,x}}\right) \quad (\text{B.12d})$$





C

## INERTIA TENSORS

*This appendix describes how the inertia tensors of the NBT are determined.*

When describing the mass distribution of a rigid body, which is free to move in three dimensions, the inertia tensor is used because it is a scalar moment of inertia of an object. The inertia tensor of a rigid body is described as:

$${}^A I = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix} \quad (\text{C.1})$$

$$I_{xx} = \iiint_V (y^2 + z^2) \rho(x, y, z) dv$$

$$I_{yy} = \iiint_V (x^2 + z^2) \rho(x, y, z) dv$$

$$I_{zz} = \iiint_V (x^2 + y^2) \rho(x, y, z) dv$$

$$I_{xy} = \iiint_V xy \rho(x, y, z) dv$$

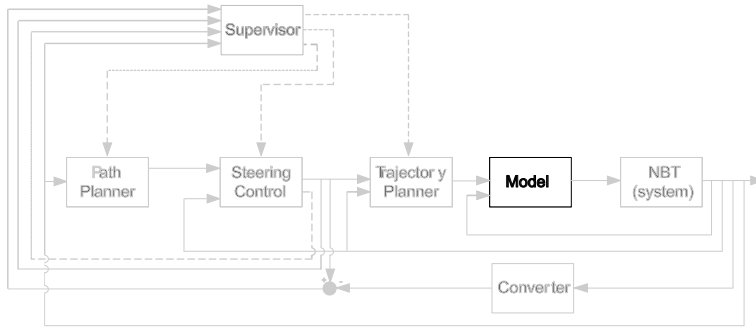
$$I_{xz} = \iiint_V xz \rho(x, y, z) dv$$

$$I_{yz} = \iiint_V yz \rho(x, y, z) dv$$

Where  $\rho(x, y, z)$  is the mass density at the specific point  $(x, y, z)$ . To simplify the model of the NBT, it is divided into small parts which are described as boxes or plates. These parts will be assumed to have a homogeneous mass density distribution, which means that  $\rho(x, y, z)$  will be constant for any point  $(x, y, z)$  within the object boundaries.

The inertia tensor is calculated in the following MATLAB<sup>®</sup> file `/simulation/model/Inertia_tensor.m` This appendix is building on [Cra05, section 6.3 p. 190-195].





D

## MANIPULATOR DYNAMICS

*This appendix concerns extra stuff for the dynamics of the NBT derived in Chapter 11 on page 61. The gravity force will be divided between the supporting legs, the contents of the state-space equations derived in Equation (D.5) will be shown and the state-space equations will be structurally tested. The contents of the state-space equations will only be shown for the LF leg because the contents are akin for all the legs.*

### D.1 DISTRIBUTION OF GRAVITY FORCE

In construction of buildings and bridges it is important to calculate the force on the support points to secure that they are strong enough to hold the construction. The calculation of the force on the supporting points can be used for calculating the gravity force in the shoulders. This is illustrated on Figure D.1(a) for the NBT, The view direction is illustrated on Figure D.1(b). Figure D.1(a) is only illustrating the supporting legs and the body, seen as the line between the supporting legs. An equivalent diagram for Figure D.1(a) is illustrated on Figure D.1(c), as a point load diagram.

The body can have an angle to the floor, as described in Chapter 4 on page 17, this is illustrated on Figure D.2(a). This is equivalent to a line load diagram as illustrated on Figure D.2(b), which is equivalent to the point load diagram as illustrated on Figure D.1(c).

The angle of the body do not have influence on how the force is divided out to the shoulders because the line load is equal to the point load. The calculation of the force in the shoulders depend on the number of supporting legs, it can have two or four supporting legs. The two scenarios is described in the following:

**Two supporting legs:** When the NBT has two legs on the ground the gravity force,  $F_g$ , is divided between them. The location of the CM gives the ratio for the splitting of  $F_g$ , these are illustrated on Figure D.3.

$L_A$  and  $L_B$  are the supporting legs and  $L_{CM \rightarrow L_A}$  and  $L_{CM \rightarrow L_B}$  are the lengths from

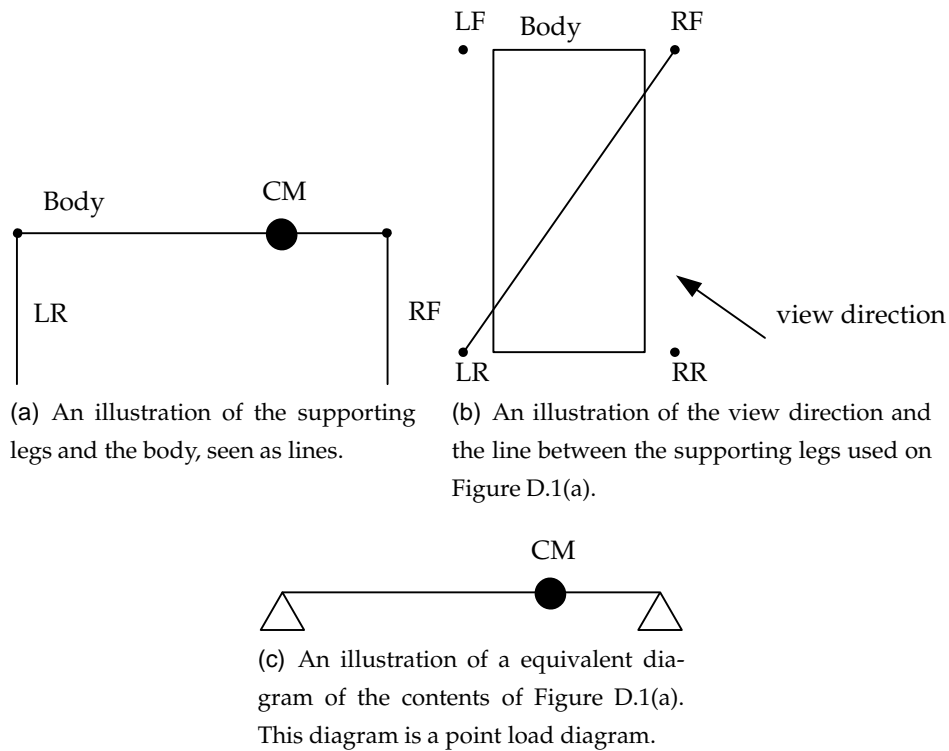


FIGURE D.1: Illustrations of how the support can be seen.

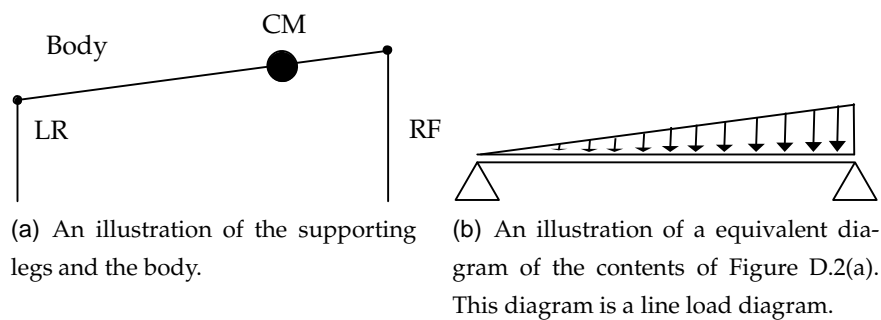


FIGURE D.2: Illustrations of how the support can be seen when the body is in an angle to the floor.

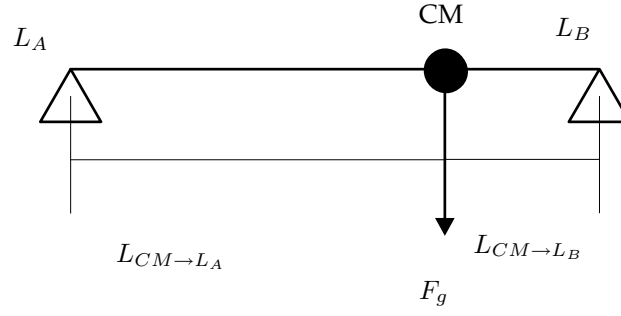


FIGURE D.3: Illustration of the point load diagram with two supporting legs, with the lengths to the legs illustrated.

the CM to  $L_A$  and  $L_B$ , respectively. The distances from the CM to the shoulders shall be measured parallel to the floor in the calculations of the force in the shoulders. The gravity force in the shoulders are calculated in Equation (D.1).

$$F_{g,A} = \frac{L_{CM \rightarrow L_B}}{L_{CM \rightarrow L_A} + L_{CM \rightarrow L_B}} F_g \quad (D.1a)$$

$$F_{g,B} = \frac{L_{CM \rightarrow L_A}}{L_{CM \rightarrow L_A} + L_{CM \rightarrow L_B}} F_g \quad (D.1b)$$

$F_{g,A}$  and  $F_{g,B}$  are the gravity force in  $L_A$  and  $L_B$ , respectively, given in the plane parallel with the floor. If the CM not is on the line between the supporting legs will the gravity start an angle acceleration around the feet of the supporting legs, as described in Chapter 4 on page 17.

The formulas in Equation (D.1) is checked for if it gives  $F_g$  in Equation (D.2).

$$\begin{aligned} F_g &= F_{g,A} + F_{g,B} \\ &= \frac{L_{CM \rightarrow L_B}}{L_{CM \rightarrow L_A} + L_{CM \rightarrow L_B}} F_g + \frac{L_{CM \rightarrow L_A}}{L_{CM \rightarrow L_A} + L_{CM \rightarrow L_B}} F_g \\ &= \frac{L_{CM \rightarrow L_A} + L_{CM \rightarrow L_B}}{L_{CM \rightarrow L_A} + L_{CM \rightarrow L_B}} F_g \\ &= F_g \end{aligned} \quad (D.2)$$

**Four supporting legs:** The calculations of the gravity force in the shoulders when supported by four legs is akin to the calculations when supported by two legs,  $F_g$  is just divided one more time. The lengths from the CM are illustrated on Figure D.4.

$L_{CM \rightarrow L}$ ,  $L_{CM \rightarrow R}$ ,  $L_{CM \rightarrow F}$  and  $L_{CM \rightarrow B}$  are the length from the CM to left-, right side, front and back respectively. The gravity force in the shoulders are calculated

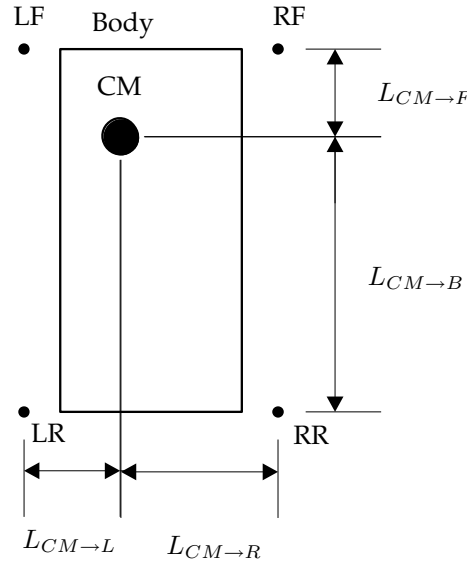


FIGURE D.4: Illustration of the lengths from the CM to the lines between two neighbouring legs.

in Equation (D.3).

$$F_{g,LF} = \frac{L_{CM \rightarrow R}}{L_{CM \rightarrow L} + L_{CM \rightarrow R}} \frac{L_{CM \rightarrow R}}{L_{CM \rightarrow F} + L_{CM \rightarrow B}} F_g \quad (D.3a)$$

$$F_{g,RF} = \frac{L_{CM \rightarrow L}}{L_{CM \rightarrow L} + L_{CM \rightarrow R}} \frac{L_{CM \rightarrow R}}{L_{CM \rightarrow F} + L_{CM \rightarrow B}} F_g \quad (D.3b)$$

$$F_{g,LR} = \frac{L_{CM \rightarrow R}}{L_{CM \rightarrow L} + L_{CM \rightarrow R}} \frac{L_{CM \rightarrow F}}{L_{CM \rightarrow F} + L_{CM \rightarrow B}} F_g \quad (D.3c)$$

$$F_{g,RR} = \frac{L_{CM \rightarrow L}}{L_{CM \rightarrow L} + L_{CM \rightarrow R}} \frac{L_{CM \rightarrow F}}{L_{CM \rightarrow F} + L_{CM \rightarrow B}} F_g \quad (D.3d)$$

$F_{g,LF}$ ,  $F_{g,RF}$ ,  $F_{g,LR}$  and  $F_{g,RR}$  are the gravity force in the LF-, RF-, LR- and RR leg, respectively, given in the plane parallel with the floor.

The formulas in Equation (D.3) is checked for if it gives  $F_g$  in Equation (D.4).

$$\begin{aligned} F_g &= F_{g,LF} + F_{g,RF} + F_{g,LR} + F_{g,RR} \\ &= F_g \end{aligned} \quad (D.4)$$

(Note: The dividing of  $F_g$  is implemented in a function, which can be found in /simulations/model/GravityDeviding.m.)

The resulting normal forces from the floor are converted to be given in  $\{LF4\}$ ,  $\{RF4\}$ ,  $\{LR4\}$  and  $\{RR4\}$  for the given leg.

The torques for the joints will be represented as a state-space equations for each leg, these can be found in Equation (D.5). [Cra05, p. 190-191]

$$\bar{\tau}_{LF} = \bar{M}_{LF}(\bar{\theta}_{LF}) \bar{\ddot{\theta}}_{LF} + \bar{V}_{LF}(\bar{\theta}_{LF} \bar{\dot{\theta}}_{LF}) + \bar{G}_{LF}(\bar{\theta}_{LF}) \quad (\text{D.5a})$$

$$\bar{\tau}_{RF} = \bar{M}_{RF}(\bar{\theta}_{RF}) \bar{\ddot{\theta}}_{RF} + \bar{V}_{RF}(\bar{\theta}_{RF} \bar{\dot{\theta}}_{RF}) + \bar{G}_{RF}(\bar{\theta}_{RF}) \quad (\text{D.5b})$$

$$\bar{\tau}_{LR} = \bar{M}_{LR}(\bar{\theta}_{LR}) \bar{\ddot{\theta}}_{LR} + \bar{V}_{LR}(\bar{\theta}_{LR} \bar{\dot{\theta}}_{LR}) + \bar{G}_{LR}(\bar{\theta}_{LR}) \quad (\text{D.5c})$$

$$\bar{\tau}_{RR} = \bar{M}_{RR}(\bar{\theta}_{RR}) \bar{\ddot{\theta}}_{RR} + \bar{V}_{RR}(\bar{\theta}_{RR} \bar{\dot{\theta}}_{RR}) + \bar{G}_{RR}(\bar{\theta}_{RR}) \quad (\text{D.5d})$$

Where the  $\bar{M}(\bar{\theta})$  are the  $3 \times 3$  mass matrices for the manipulators,  $\bar{V}(\bar{\theta} \bar{\dot{\theta}})$  are  $3 \times 1$  vectors of the coriolis and centrifugal terms,  $\bar{G}(\bar{\theta})$  are  $3 \times 1$  vectors of the gravity.  $\bar{\tau}$ ,  $\bar{\theta}$ ,  $\bar{\dot{\theta}}$  and  $\bar{\ddot{\theta}}$  are  $3 \times 1$  vectors for the torques, joint angles, angular velocity and angular acceleration, respectively. The value of all the variables in Equation (D.5) can be calculated in `/simulations/model/dynamics_formula.m` and a simplified version can be found in Appendix D on page 145. The torque calculations are implemented as a function for each leg. (Note: The calculation of the torque for the joints can be found in `/simulations/model/dynamicsLF.m`, `/simulations/model/dynamicsRF.m`, `/simulations/model/dynamicsLR.m` and `/simulations/model/dynamicsRR.m`)

## D.2 STATE-SPACE EQUATIONS

This section concerns the contents of the state-space equations derived in Chapter 11 on page 61. The results demonstrated in this section are rounded values, the exact value of all the variables can be calculated

in `/simulations/model/dynamics_formula.m` The state space equation that will be described is shown in Equation (D.6).

$$\bar{\tau}_{LF} = \bar{M}_{LF}(\bar{\theta}_{LF}) \bar{\ddot{\theta}}_{LF} + \bar{V}_{LF}(\bar{\theta}_{LF} \bar{\dot{\theta}}_{LF}) + \bar{G}_{LF}(\bar{\theta}_{LF}) \quad (\text{D.6})$$

Equation (D.7) contain the torque vector  $\bar{\tau}_{LF}$

$$\bar{\tau}_{LF} = \begin{bmatrix} \tau_{LF1} & \tau_{LF2} & \tau_{LF3} \end{bmatrix}^T \quad (\text{D.7})$$

Equation (D.8) contains the angle vector  $\bar{\theta}_{LF}$ .

$$\bar{\theta}_{LF} = \begin{bmatrix} \theta_{LF1} & \theta_{LF2} & \theta_{LF3} \end{bmatrix}^T \quad (\text{D.8})$$

Equation (D.9) contains the angular velocity vector  $\dot{\bar{\theta}}_{LF}$ .

$$\dot{\bar{\theta}}_{LF} = \begin{bmatrix} \dot{\theta}_{LF1} & \dot{\theta}_{LF2} & \dot{\theta}_{LF3} \end{bmatrix}^T \quad (D.9)$$

Equation (D.10) contain the angular acceleration vector  $\ddot{\bar{\theta}}_{LF}$ .

$$\ddot{\bar{\theta}}_{LF} = \begin{bmatrix} \ddot{\theta}_{LF1} & \ddot{\theta}_{LF2} & \ddot{\theta}_{LF3} \end{bmatrix}^T \quad (D.10)$$

Equation (D.11) contains the angular velocity matrix  $\bar{M}_{LF}$ .

$$\bar{M}_{LF} = \begin{bmatrix} A_{M_{LF}} & B_{M_{LF}} & C_{M_{LF}} \\ D_{M_{LF}} & E_{M_{LF}} & F_{M_{LF}} \\ G_{M_{LF}} & H_{M_{LF}} & J_{M_{LF}} \end{bmatrix} \quad (D.11)$$

$$(D.12)$$

Equation (D.13) contains the angular velocity vector  $\bar{V}_{LF}$ .

$$\bar{V}_{LF} = \begin{bmatrix} A_{V_{LF}} & B_{V_{LF}} & C_{V_{LF}} \end{bmatrix}^T \quad (D.13)$$

$$(D.14)$$

Equation (D.15) contains the angular acceleration vector  $\bar{G}_{LF}$

$$\bar{G}_{LF} = \begin{bmatrix} A_{G_{LF}} & B_{G_{LF}} & C_{G_{LF}} \end{bmatrix}^T \quad (D.15)$$

$$(D.16)$$

The contents of some vectors and matrices are to large to demonstrate in this appendix, but can be found in `/simulations/model/dynamics_formula.m`

## D.3 STRUCTURAL TEST

The functions for the dynamics in Equation (D.5) are structural tested with the angles shown in Table D.1, each of these tests are performed with the velocities and accelerations in Table D.2, and for all the three standing positions.

$\theta$  is the joint angle,  $\dot{\theta}$  is the joint velocity and  $\ddot{\theta}$  is the joint acceleration. The calculated joint torques were as expected.



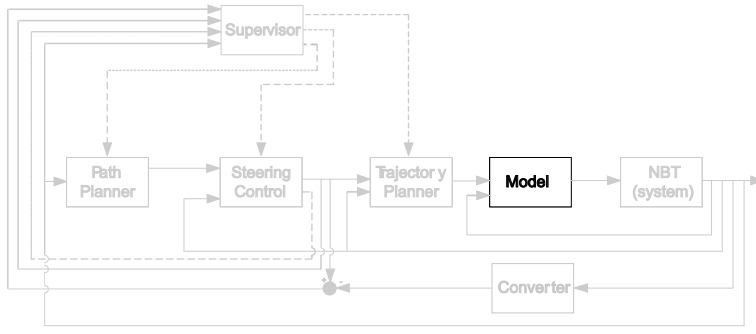
	$\theta_{LF1}$	$\theta_{LF2}$	$\theta_{LF3}$
Test 1	0 [rad]	0 [rad]	0 [rad]
Test 2	$\pi/2$ [rad]	0 [rad]	0 [rad]
Test 3	$-\pi/2$ [rad]	0 [rad]	0 [rad]
Test 4	0 [rad]	$\pi/2$ [rad]	0 [rad]
Test 5	0 [rad]	$-\pi/2$ [rad]	0 [rad]
Test 6	0 [rad]	0 [rad]	$\pi/2$ [rad]
Test 7	0 [rad]	0 [rad]	$-\pi/2$ [rad]

**TABLE D.1:** *The angles that the torques for the joints shall be calculated for in the test.*

	$\dot{\theta}_{LF1}$	$\ddot{\theta}_{LF1}$	$\dot{\theta}_{LF2}$	$\ddot{\theta}_{LF2}$	$\dot{\theta}_{LF3}$	$\ddot{\theta}_{LF3}$
Test A	0 [m/s]	0 [m/s <sup>2</sup> ]	0 [m/s]	0 [m/s <sup>2</sup> ]	0 [m/s]	0 [m/s <sup>2</sup> ]
Test B	1 [m/s]	0 [m/s <sup>2</sup> ]	0 [m/s]	0 [m/s <sup>2</sup> ]	0 [m/s]	0 [m/s <sup>2</sup> ]
Test C	0 [m/s]	1 [m/s <sup>2</sup> ]	0 [m/s]	0 [m/s <sup>2</sup> ]	0 [m/s]	0 [m/s <sup>2</sup> ]
Test D	0 [m/s]	0 [m/s <sup>2</sup> ]	1 [m/s]	0 [m/s <sup>2</sup> ]	0 [m/s]	0 [m/s <sup>2</sup> ]
Test E	0 [m/s]	0 [m/s <sup>2</sup> ]	0 [m/s]	1 [m/s <sup>2</sup> ]	0 [m/s]	0 [m/s <sup>2</sup> ]
Test F	0 [m/s]	0 [m/s <sup>2</sup> ]	0 [m/s]	0 [m/s <sup>2</sup> ]	1 [m/s]	0 [m/s <sup>2</sup> ]
Test G	0 [m/s]	0 [m/s <sup>2</sup> ]	0 [m/s]	0 [m/s <sup>2</sup> ]	0 [m/s]	1 [m/s <sup>2</sup> ]

**TABLE D.2:** *The velocities and accelerations that the torques for the joints shall be calculated for in the test.*





E

## STRUCTURAL TEST OF MODEL

*This appendix contains a test of the model. The test will be based on a structural test, investigating the overall correctness of the model and the signs used.*

Since the model parts are combined in Simulink<sup>®</sup>, see Chapter 12 on page 67, the verification is also made in Simulink<sup>®</sup>. The verification will be conducted in four different test scenarios, where the following inputs will be applied to the model:

**Test scenario 1.** Constant values for all the inputs.

**Test scenario 2.** Predefined value for  $\theta_{LF1}$  constant values for the other inputs.

**Test scenario 3.** Predefined value for  $\theta_{LF2}$  constant values for the other inputs.

**Test scenario 4.** Predefined value for  $\theta_{LF3}$  constant values for the other inputs.

The predefined input that will be used in test scenario 2 to 4 is shown in Figure E.1.

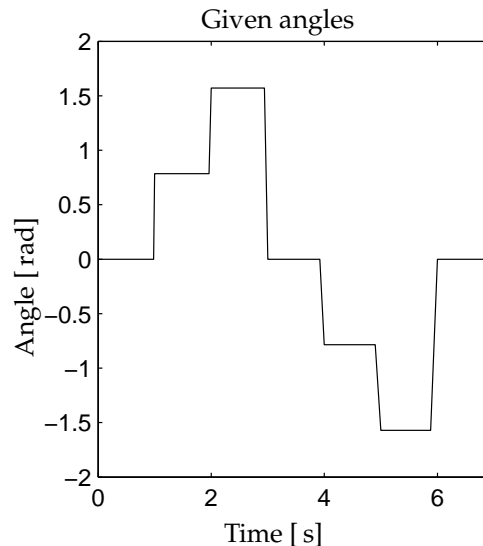


FIGURE E.1: The variable input there will be used in test scenario 2 to 4.

The figures used in this appendix show test data produced by the function, `/simulation/model/plot_verification_model.m`. Note that the figures for the calculated torque have different scales on the y axis for clarity.

The test procedures are similar for all the legs, therefore only the verification of the LF leg will be documented. If there should be differences in the results of the tests for the legs, they will be described in the end of this appendix.

The test setup for the LF leg is illustrated in Figure E.2.

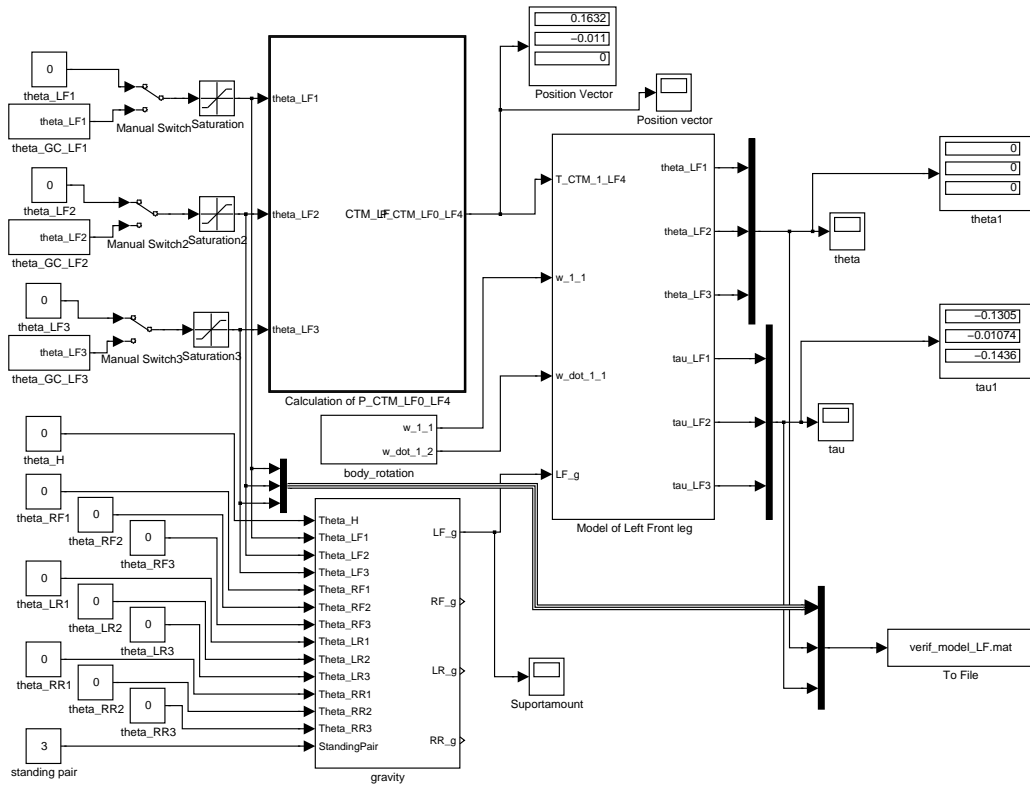


FIGURE E.2: System setup for verification of the model for the LF leg.

*Calculation of  $T_{CTM\_1\_LF4}$*  is an Embedded MATLAB<sup>®</sup> function calling the function for calculating the CTM, made in Chapter 9 on page 47. This function is calculating the transformation matrix  ${}^1_{LF4}T_{CTM}$ , from the three given angles. *Model of the Left Front leg* is the subsystem containing the model, which was described in Chapter 12 on page 67. The three *Manual Switches* change  $\theta_{LF2}$ ,  $\theta_{LF3}$  and  $\theta_{LF4}$  between constant and predefined value. The three *saturation* boxes ensure that the values do not exceed the working range of the joint defined in Chapter 10 on page 55. *Gravity* is a subsystem splitting the CM out to the supporting legs, as described in Chapter 11. *Body\_rotations* is a subsystem defining the angle and angular velocity for the body of the NBT.

## E.0.1 TEST SCENARIO 1.

**Description:** The inputs will be changed one by one, while the other inputs be hold to zero. The inputs that will be applied are  $0$ ,  $\pi/4$ ,  $\pi/2$ ,  $-\pi/4$  and  $-\pi/2$  [rad], if they are inside the working area. The specifications of the working range can as earlier mentioned be found in Chapter 10. When the legs are in the initial position are the supporting LP shifted.

**Expected result:** The angles are expected to be calculated with a precision of  $0.01$  [rad]. Changes in one input is expected to have influence on the required torque for all three joints, as the inputs are constant values. There will not be any acceleration and deceleration when changing value. The required torques are expected to be larger when the leg is supporting the body.

**Test result:** Table E.1 contains the results of the tests. Angles are donated  $\theta$  and torque are donated  $\tau$ .

Input				Output					
$\theta_{LF1}$	$\theta_{LF2}$	$\theta_{LF3}$	pair	$\theta_{LF1}$	$\theta_{LF2}$	$\theta_{LF3}$	$\tau_{LF1}$	$\tau_{LF2}$	$\tau_{LF3}$
0	0	0	LP1*	0	0	0	-0.0104	-0.0107	0.0009
0	0	0	LP2	0	0	0	-0.1245	-0.0107	-0.1436
0	0	0	LP3	0	0	0	-0.1305	-0.0107	-0.1436
$\pi/4$	0	0	LP1	0.7854	0	0	0.1560	-0.0107	0.0267
$\pi/2$	0	0	LP1	1.571	0	0	0.2103	-0.0107	0.0369
$-\pi/4$	0	0	LP1	-0.7854	0	0	-0.1414	-0.0107	-0.0254
$-\pi/2$	0	0	LP1	-1.571	0	0	-0.2103	-0.0107	-0.0369
0	$\pi/4$	0	LP1	0	0.7854	0	0.0083	0.0818	0.0012
0	$-\pi/4$	0	LP1	0	-0.7854	0	0.0083	-0.0970	0.0013
0	$-\pi/2$	0	LP1	0	-1.571	0	0.0035	-0.1264	0.0021
0	0	$\pi/4$	LP1	0	0	0.7854	0.0362	-0.0099	0.0267
0	0	$\pi/2$	LP1	0	0	1.571	0.0463	-0.0077	0.0369

**TABLE E.1:** Results from applying constant values as joint angles for the LF leg. \* When the angles for the leg joints all are zero is the leg in a position where it is supporting the weight of the body, but to test the difference in torque when supporting the weight and not supporting the weight is this test included.

**Test conclusion:** The given angles for the inputs  $\pm\pi/2$  and  $\pm\pi/4$  are calculated to  $\pm 1.571$  and  $\pm 0.7854$ , respectively, which are correct. Therefore the precision for the calcula-

---

tion of the angles is at least 0.001 which fulfill the expected precision. The required torque are changing when the angles are changing, and the changes in the angle are also influencing the required torque for the other joints as expected. The required torques are as expected larger when the leg is supporting the weight of the body, which is for LP2 and LP3.

## E.0.2 TEST SCENARIO 2.

**Description:**  $\theta_{LF1}$  will be applied as the signal shown in figure E.1 while the two other inputs will be a constant zero.

**Expected result:** The angles are expected to be calculated such that they follow their given input. Changes in  $\theta_{LF1}$  is expected to give an increase in the required torque, which will accelerate the leg, and make it start moving. This will be followed by a decrease in the required torque, which will decelerate the leg and stop it at a certain position. The torque at this level is the torque required to hold the angle  $\theta_{LF1}$ . The required torque is expected to be largest for  $\{LF1\}$  because this is the first joint. The torque for  $\tau_{LF2}$  is only expected to be influenced slightly when  $\theta_{LF1}$  is changed because it is perpendicular to the rotation direction.  $\{LF3\}$  is expected to have similar changes as  $\{LF1\}$ , just smaller.

**Test result:** Figure E.3 illustrated the results of the test. The blue line is for  $\{LF1\}$ , the magenta line is for  $\{LF2\}$  and the black line is for  $\{LF3\}$ . This notation will be used through out this appendix.

**Test conclusion:** The calculated angles for  $\theta_{LF1}$  follows the given input and the calculated angles for the other two inputs is a constant zero as expected. When  $\theta_{LF1}$  is changing the torque for  $\tau_{LF1}$  increases then decreases and finally returns to a level where it is stable. This level depends on the given angle, as it should. Changes in  $\tau_{LF2}$  are only minor and  $\tau_{LF3}$  follows  $\tau_{LF1}$  as expected.

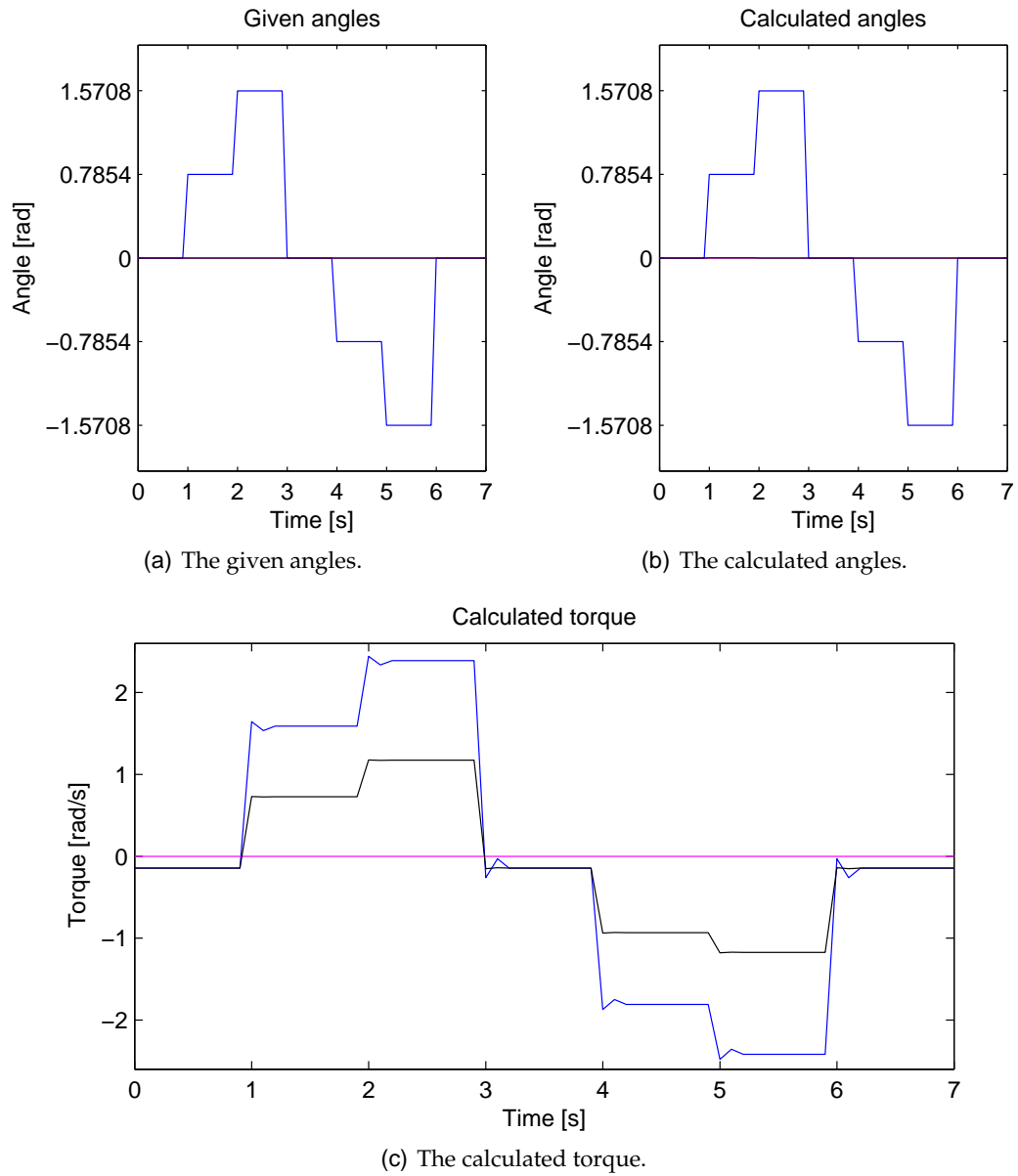


FIGURE E.3: Test results for test scenario 2 for the LF leg, from applying a variation values to  $\theta_{LF1}$ . The blue line is for {LF1}, the magenta line is for {LF2} and the black line is for {LF3}.

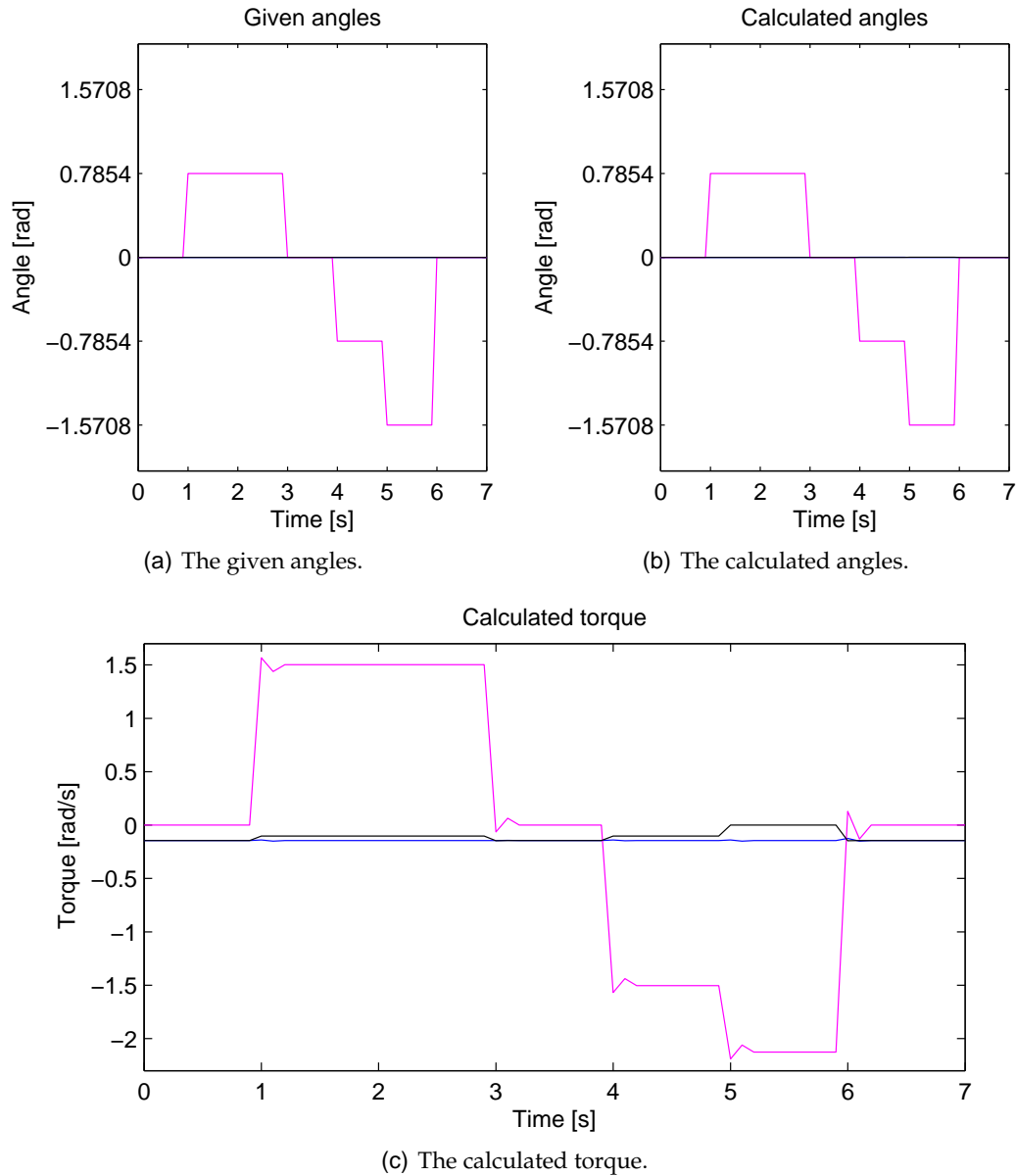
### E.0.3 TEST SCENARIO 3.

**Description:**  $\theta_{LF2}$  will have the signal shown in Figure E.1 applied, the two other inputs will be a constant zero.

**Expected result:** The given and calculated angles are expected to follow each other and

the changes in  $\theta_{LF2}$  are expected to affect the required torque for all three joints in the leg. The changes in the torque will make the accelerations as described in section E.0.2.

**Test result:** Figure E.4 shows the results of the test.



**FIGURE E.4:** Test results for test scenario 3 for the LF leg, from applying a variation values to  $\theta_{LF2}$ . The blue line is for {LF1}, the magenta line is for {LF2} and the black line is for {LF3}.

**Test conclusion:** The calculation of all the angles follow their given input. When  $\theta_{LF2}$  changes the required torque for all three joints also changes with acceleration and



deceleration as expected. The changes in level for  $\tau_{LF3}$  are difficult to see, but do exist. The changes in level for  $\tau_{LF1}$  are also difficult to see, but they are also there.

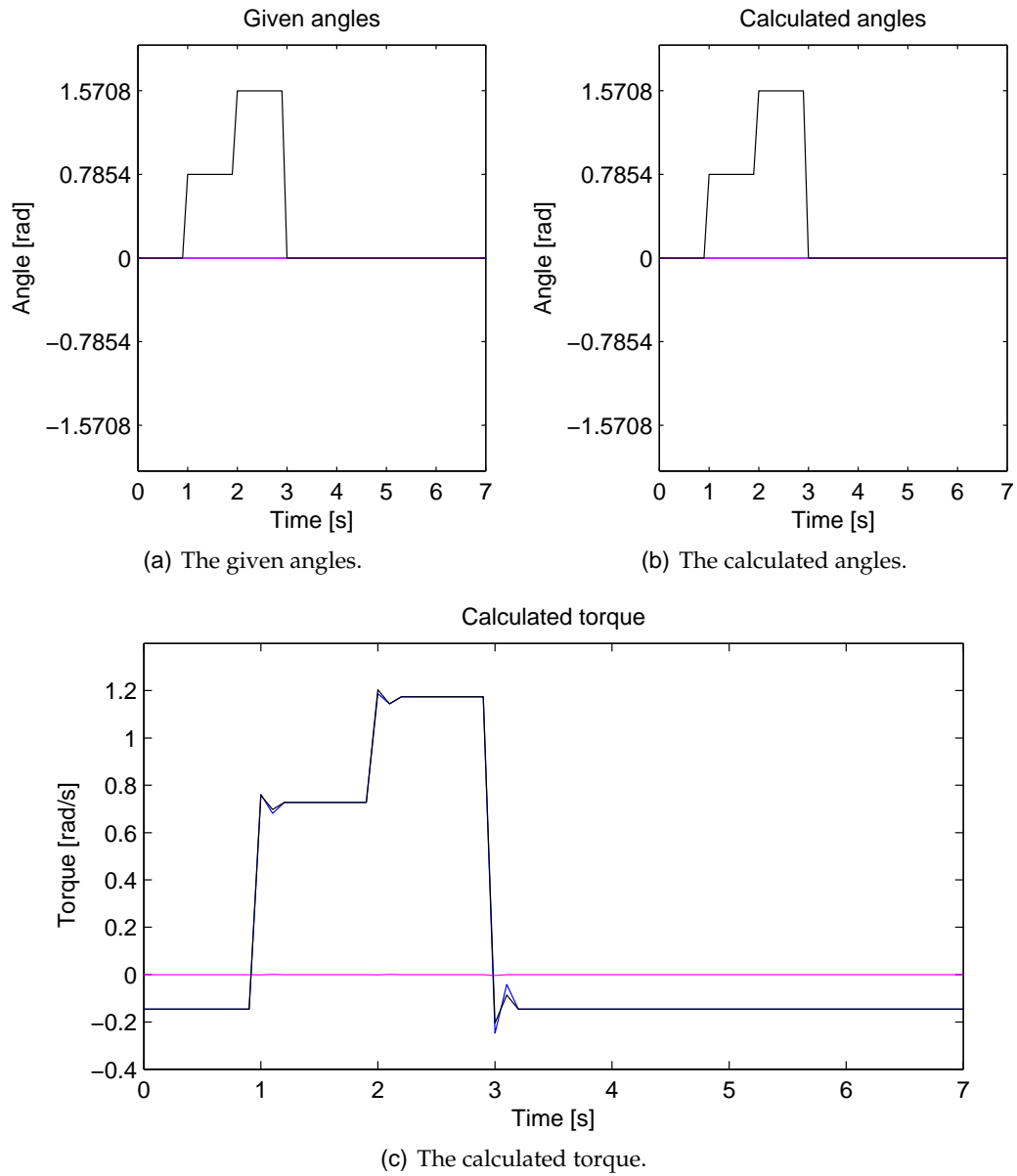
#### E.0.4 TEST SCENARIO 4.

**Description:**  $\theta_{LF3}$  will have the signal shown in Figure E.1 applied and the two other inputs will be a constant zero.

**Expected result:** The result should be akin to the result from section E.0.2, The difference is that the torque will not change as much, because the mass of the lower leg is smaller than the mass of the upper leg.

**Test result:** Figure E.5 illustrates the results of the test.

**Test conclusion:** The calculated angles and torques are akin to the calculated torque in section E.0.2, the difference being that  $\tau_{LF2}$  is smaller. The joint is not rotated in negativ direction because this is outside the working range. This was the expected result.



**FIGURE E.5:** Test results for test scenario 4 for the LF leg, from applying a variation values to  $\theta_{LF3}$ . The blue line is for  $\{LF1\}$ , the magenta line is for  $\{LF2\}$  and the black line is for  $\{LF3\}$ .